VIVA-TECH IJRI
ANNUAL RESEARCH JOURNAL
VIVA-TECH INTERNATIONAL JOURNAL FOR RESEARCH AND INNOVATION
ANNUAL RESEARCH JOURNAL
ISSN(ONLINE): 2581-7280

A study on Interplanetary File System based on Blockchain

Aditi Dilip Kudu[1], Harshada Shantilal Prajapati[2]
[1](MCA, VIVA Institute of Technology / University of Mumbai, India)
[2](MCA, VIVA Institute of Technology / University of Mumbai, India)

Abstract : *The Interplanetary File System (IPFS) is a distributed file system that seeks to decentralize the web and to make it faster and more efficient. It incorporates well- known technologies, including BitTorrent and Git, to create a swarm of computing systems that share information. Since its introduction in 2016, IPFS has seen great improvements and adoption from both individuals and enterprise organizations. Its distributed network allows users to share files and information across the globe. IPFS works well with large files that may consume or require large bandwidth to upload and/or download over the Internet. The rapid adoption of this distributed file system is in part because IPFS is designed to operate on top of different protocols, such as FTP and HTTP. However, there are underpinning concerns relating to security and access control, for example lack of traceability on how the files are accessed. The aim of this paper is to complement IPFS with blockchain technology, by proposing a new approach (BlockIPFS) to create a clear audit trail. BlockIPFS allows us to achieve improved trustworthiness of the data and authorship protection, and provide a clear route to trace back all activities associated with a given file using blockchain as a service.*

Keywords- *BlockIPFS, HTTP, IPFS, FTP, blockchain, etc.*

I. INTRODUCTION

The IPFS is a peer-to-peer distributed file system that seeks to connect all computing devices with the same system of files. The conveyed network, joining advances like BitTorrent and Git, accomplishes high throughput and permits clients to share records and data productively across the organization. IPFS network is ideal for sharing enormous records that may burn-through or require huge data transfer capacity to transfer as well as download over the Internet, and it has been intended to work on top of different protocols. The central idea that developers built IPFS on is to model all data as part of the same Merkle DAG [1] Nonetheless, the prominence and viability of IPFS as a conveyed record framework likewise make security and access control concerns. In a circulated organization like IPFS, when an item is transferred to the organization, any individual who approaches the hash address of the document can get to its substance [1]. Albeit this is a helpful component for an appropriated record framework, the client who transfers a document can't handle admittance to the transferred record once the hash address has been shared.

Cryptographic hashes that serve to securely identify a file's content, can be sent to the latter, thus proving that the file was available to someone at a certain time. One particularly interesting file sharing platform for this purpose, combining file sharing and the mentioned hashes, is the Inter Planetary File System (IPFS) [2]. IPFS identifies, verifies and transfers files relying on the cryptographic hashes of their contents. Similar to public blockchains, files stored on ipfs can be requested and viewed by anyone who can connect to or deploy an ipfs node. This is an issue for blockchain applications working with large files that contain sensitive or personal data. Consequently, this paper use the Hyperledger blockchain [2] to give an entrance controlled IPFS. A chaincode stores and allows dynamic modification of the access control list. The modified IPFS software, hereinafter named BlockIPFS.

The focus of this work is to address the traceability problem so that all activities concerning a specific object on a distributed file system can be traced and access controlled. Indeed, even on a private IPFS organization, clients may wish to restrict access of a document they transferred to the organization. In the current IPFS, it is basically impossible for an association to make such a standard for getting to documents on their private IPFS

organization. In the event that a document intended to be divided between just significant level administration is shared on the organization, any individual from the association's IPFS network with admittance to the CID of the record can simply access it. That being said, the IPFS gives no discernibility abilities to record and review admittance to documents on the organization

Blockchain is a decentralized information the board stage that gives unchanging nature [4]; thus, it is a solid match to help record discernibility metadata in a dispersed document framework like IPFS. There are by and large two classes of blockchains: permissionless and permissioned. A permissionless blockchain (e.g., Ethereum, Bitcoin) is open to the public, and every transaction is to be validated by every or majority of participants [5].

II. LITERATURE REVIEW

According to J. Benet [2], IPFS is a decentralized file-sharing platform that identifies files through their content. It relies on a distributed hash table (DHT) to retrieve file locations and node connectivity information. When a file is getting uploaded to IPFS then that file gets divided into chunks. Chunks are the smaller unit of blocks.

Each chunk has a limit of data storage. Each chunk has a unique content identifier. Markle-directed acyclic graph (Markle DAG) is responsible for reconstructing any file from particular chunks. In IPFS only a hash is required to access any file or chunk. Here we cannot create a file with the same hash.

Storing and retrieving the files or folders by using blockchain is a very hard task in some cases. Just consider large data files. Storing large files on the blockchain is not possible as large file data get split into several blocks. Hence assembling this data requires a different system or additional data. Here another important thing is mining the nodes. A huge amount of data must have to propagate through these nodes. Now processing and storing this whole data is also very difficult. Since mining of nodes requires higher bandwidth so we can say that blockchain is not the perfect way to store and share large files.

Data storing on Ethereum blockchain is unchangeable but for storing the data after certain limit has cost model which is differ from conventional data storage. According to S. Nakamoto [6] bitcoin provides OP_RETURN opcode for storing the data for transactions. Before February 2014 the limit for storing transaction is 80 bytes but after that it was reduced to 40 bytes. This Ethereum blockchain requires gas fees for storage. According to Xu, I. Weber, M. Staples, L. Zhu, J. Bosch, L. Bass,C. Pautasso, and P. Rimba [7] for storing 80 bytes in bitcoin blockchain the cost is nearly US$0.03617 and US$0.007.

As it is clear that only Blockchain is not useful for sharing and storing large files hence file shearing platform can be leveraged to support the application. Due to this file system, now cryptographic hashes are responsible to get the content of a file very efficiently at a time on different nodes. Here is some data storage solution which is friendly to blockchain such as, Storj, FileCoin, Sia, IPFS. Here IPFS is the Interplanetary file system that is used with Ethereum blockchain to overcome various drawbacks of it.

III. PROBLEM DEFINATION

A circulated document framework additionally makes security and access control concerns. In a circulated organization like IPFS, when an item is transferred to the organization, any individual who approaches the hash address of the record can get to its content Albeit this is a beneficial element for a circulated document framework, the client who transfers a record can't handle admittance to the transferred document once the hash address has been shared.

The focus of this work is to address the authentication and traceability problem so that all activities concerning a specific object on an IPFS can be traced and access controlled. Even on a private IPFS network, users may wish to limit access of files they upload to the network. If a CID meant to be shared among only high-level management is shared on the IPFS network, any member of the organization's IPFS network with access to the hash of the file can simply access it. That being said, the IPFS gives no discernibility capacities to record and review admittance to documents on the organization.

IV. PROPOSED SYSTEM

Architecture of BlockIPFS:

IPFS is regularly utilized as a capacity stage for information sharing. It has the advantages of high availability and good performance, but lacks the capability of tracing access and authentication, which makes it difficult to investigate unauthorized access and authorship. In this plan, a blockchain is incorporated in IPFS. File operations such as adding or accessing a file generates metadata that is logged on a Hyperledger Fabric blockchain. The blockchain is responsible for storing and managing the metadata of the file. Figure shows an

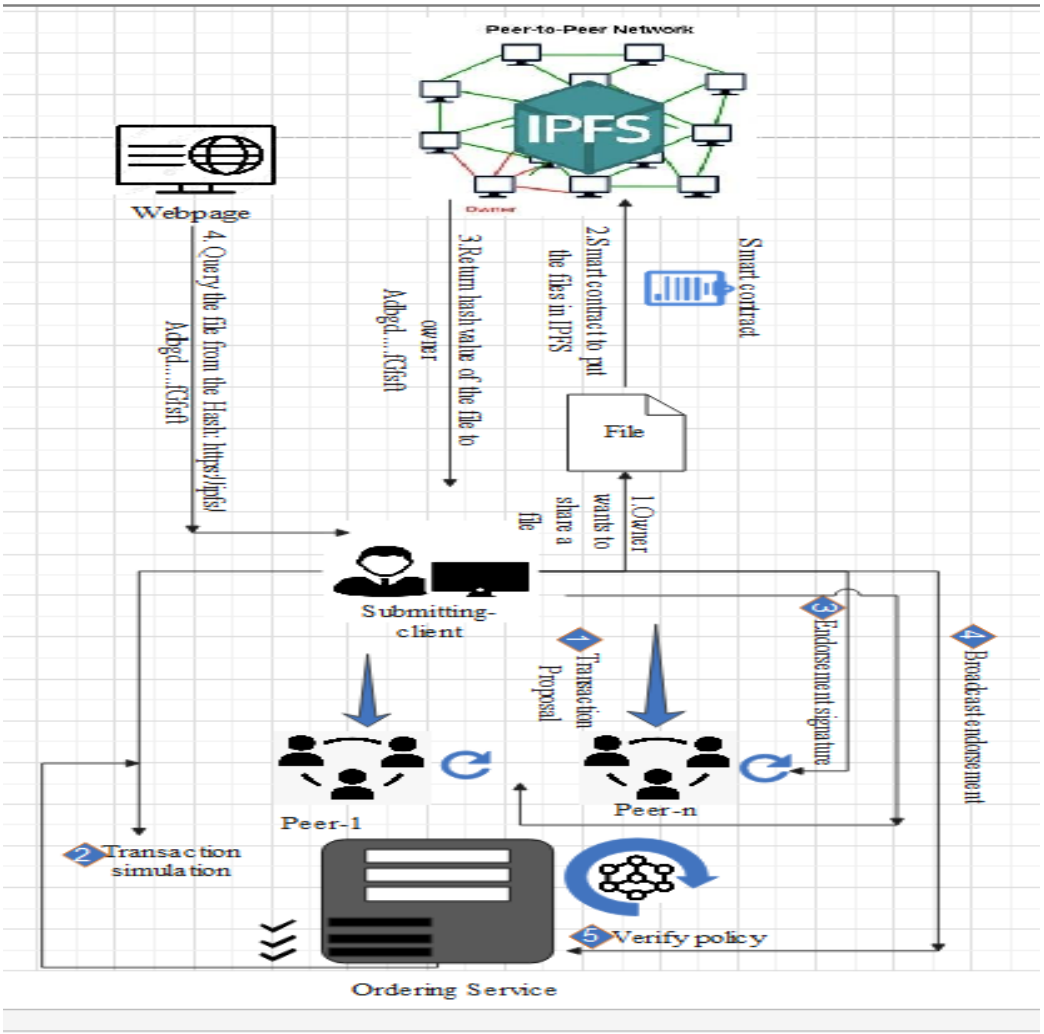
outline of the engineering of BlockIPFS, where the client can play out all recognizability activities by questioning the texture record through the client.

In BlockIPFS, the client collaborates with IPFS and can play out every one of the tasks accessible to him/her. In this implementation, when a user accesses file on other nodes, no records are pushed to their local blockchain. However, access logs are recorded in the file owner's BlockIPFS. The client can recover metadata from their nearby BlockIPFS to follow exercises identified with a particular record as well as all documents they have added to IPFS.

It worth noting that the Hyperledger Fabric blockchain in BlockIPFS only stores the metadata of the files for tracing purposes, while the files themselves are still managed by IPFS. All in all, the blockchain is practically straightforward to the clients when they read/compose documents, and any record access control instruments in IPFS will be acquired by BlockIPFS. The users only query the blockchain to exam the metadata of a file, and such metadata is recorded on the blockchain ledger in an encrypted manner so that just the proprietor of the document or the allowed clients will actually want to peruse.

V. METHODOLOGY

Fig 1.1 Transaction Flow Diagram



The fig 1.1 shows how transaction is flowing in the BlockIPFS. The process includes two phases first phase starts with the IPFS the user interacts with the file system to convert the file into the hash.

Phase 1:

User use the companion app to get the access to the local node using a web browser. This app support peer type file sharing. The app initializes and run the daemon process on local machine 127.0.0.1. it launches an API server on port 5001 and a getaway on TCP port 8080.
User start the file conversation by adding the file to the IPFS network. IPFS uses SHA256 algorithm to convert the file content into the specific length hash of 256 byte and use the Markel
DAG to track the file and return the hash to the application. This hash is the use for the further process in the next phase.

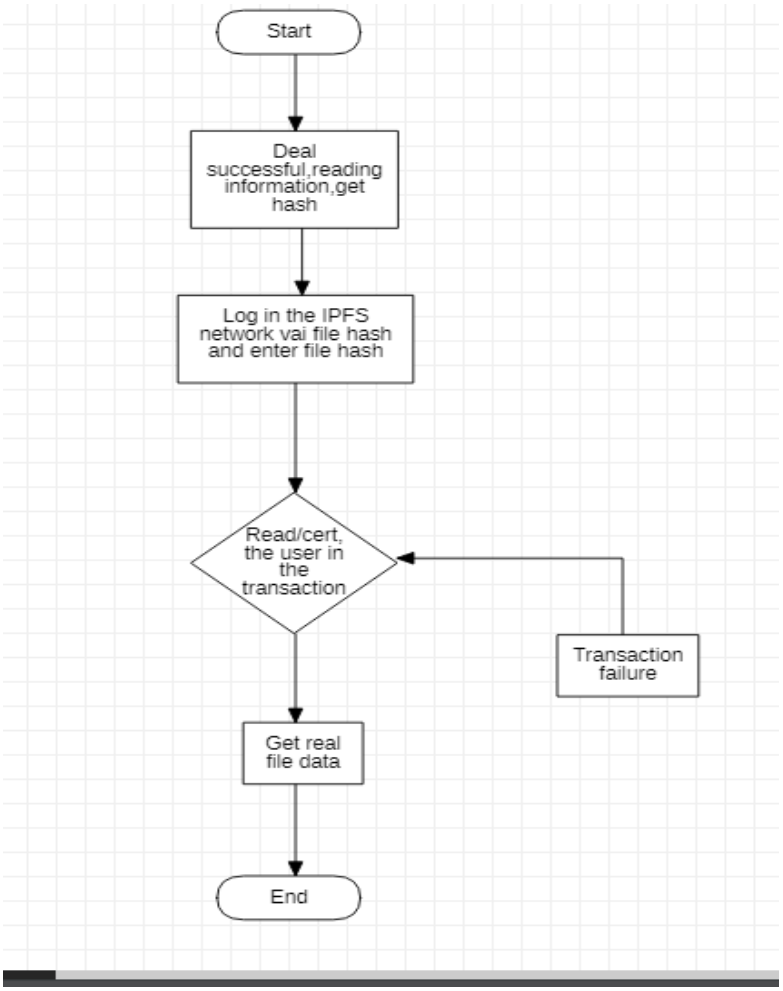


Fig.1.2 : IPFS network flow chart

Phase 2:
In phase 2 we are assuming that a channel is set up and running. The application client has enlisted and selected with the association's Certificate Authority (CA) and get back essential cryptographic material, which is utilized to validate to the organization. The chaincode is installed on peers and deployed to the channel. An endorsement policy has also been set for this chaincode, stating that both peers must endorse any transition.

1.Transation proposal:

When client send the request to add file hash. This request targets peers, next the transaction proposal is constructed. Application leveraging a SDK which use available API to generate a transaction proposal. The proposition is a solicitation to summon a chaincode furcation with certain info boundaries, with the goal of perusing the ledger.

The SDK use shim to bundle the exchange proposition into the appropriately architected arrangement and takes the client's cryptographic certifications to create an interesting mark for this change proposition.

2.Transation simulation:

At beginning with the endorsing peer verify the transition proposal and then it check the transition is already submitted or not in the past and the signature is valid or not, then the transition proposal also check for is properly authorized to perform the proposed operation on the channel.

The working of endorsing peer starts with, it takes transition proposal as input to invoke the chaincode function. The chaincode is then executed against the current state to produce transaction results including a response value, read set, and write set.

3.Endorsement signatures:

The application confirms the embracing peer marks and analyzes the proposition reactions to decide whether the proposition reactions are something very similar. On the off chance that the chaincode is just questioning the record, the application would just examine the inquiry reaction and would commonly not present the exchange to the requesting administration. On the off chance that the customer application plans to present the exchange to the requesting administration to refresh the record, the application decides whether the predefined underwriting strategy has been satisfied prior to submitting The design is to such an extent that regardless of whether an application decides not to investigate reactions or in any case advances an unendorsed exchange, the underwriting strategy will in any case be implemented by peers and maintained at the submit approval stage.

4. Broadcasting Endorsement:

The application " broadcasts " the transaction proposition and reaction inside a " transaction message" to the requesting ordering service. The transaction will contain the read/compose sets, the endorsing peers' marks and the Channel ID. The requesting ordering service doesn't have to assess the whole substance of a transaction to play out its activity, it just gets transaction from all directs in the organization, orders them sequentially by channel, and makes squares of exchanges per channel.

5. Verify policy:

The squares of exchanges are "conveyed" to all companions on the channel. The exchanges inside the square are approved to guarantee underwriting strategy is satisfied and to guarantee that there have been no progressions to record state for read set factors since the read set was produced by the exchange execution. Exchanges in the square are labelled as being substantial or invalid.

Each friend adds the square to the channel's chain, and for each legitimate exchange the compose sets are focused on present status data set. An occasion is radiated by each friend to tell the customer application that the exchange (summon) has been permanently added to the chain, just as notice of whether the transaction was validated or invalidated.

6.1 Working procedure:

The working procedure of the BlockIPFS is explained below. The following are the different steps included in BlockIPFS.

1. 1.Connect the ipfs network
2. 2.Use any ipfs getaway or client to generate the hash of the file.
    a. 3.Get the output of the file hash.
3. 4.Check the fabric network is running and chaincode is set.
4. 5.Instanshat the chaincode.
5. 6.Query the chaincode with different functions.

When end-user wants to upload a file or director to ipfs it executes the "ipfs add <file_name>" command, there are many ways to do this like using ipfs gateways and client. Internal this command calls the hash algorithm SHA256 which convert input data to fix value of 256-bit value hash.

Now end-user interacts to fabric network to authentication and secure the hash (file). If user is not register to network it has to register first for registration user has follow the two steps:

It has to provide user name and role (by default role is user) this step gives the secrete key.

Enrols the user by using the user's name and the generated secrete key.

After the registration user can use the network to invoke the transition and add the data in ledger. The data is added to word state of the ledger and all the meta-data word state is added to the transition logs.to get the data user has to query the ledger it will gives the user data back to user. User can track the data by using transaction logs.

When user get or send the file hash to any other user, they have interacted with ipfs by using one of the anther way getaway or the client. They execute the command "ipfs get<file_hash>" it will download the file in user's system. Internally this command finds the Merkle DAG and Get a raw IPFS block.

VI. CONCLUSION

The system addressed the requirement of blockchain applications to share files containing sensitive information. As discussed, the files can neither be efficiently stored on the blockchain nor be uploaded through unmodified IPFS nodes. For this purpose, the design and implementation of BlockIPFS, a blockchain-based extension to IPFS that provides access control, have been discussed. BlockIPFS leverages fabric smart contracts to handle access. Through the smart contract, users can register files, and grant or revoke access to them. To this end modified IPFS software provides access to the smart contract and enforces the permissions stored in the fabric leger. As the number of nodes is less it enhances the scalability of the system.

ACKNOWLEDGMENTS

We would like to express our sincere thanks to our guide Prof. Krutika Vartak for taking time from her busy schedule to provide us with a great deal of help, support and encourage us to work diligently at every aspect of our research paper. Her views have always been equitably providing a perfect balance between encouragement and constructive criticism. Her tips and suggestions helped us to decide the correct approach to the research paper. We attempted to find help from a variety of individuals at various stages of the research paper. We would like to thank everyone for their guidance.

REFERENCES

[1] Dr. Christian Lundkvist *and John Lilic*, what does an IPFS file look like? How should we build it? https://www.biyungu.com/news/2674.html
[2] Kimberley Mok, Interplanetary File System Could Pave the Way for a Distributed, Permanent Web, https://thenewstack.io/interplanetary-file-system-could- pave-the-way-for-a-distributed-permanent-web/
[3] Justin Johnson, (October 7, 2016), What is the interplanetary File System?, https://blog.stackpath.com/glossary-ipfs/
[4] Official Documentation of IPFS, https://ipfs.io/ Investopedia. 2021. Investopedia. [online] Available at: <http://www.investopedia.com/> [Accessed 4 July 2021].
[5] Laurence, T., 2017. Blockchain for Dummies. 1st ed. Wiley & Sons Canada, Limited, John, p.56.
[6] By Rahul Venugopal, Simplilearn.com. 2021. World's #1 Online Bootcamp & Certification Course Provider | Simplilearn. [online] Available at: <https://www.simplilearn.com/> [Accessed 3 July 2021].
[7] Jake Frankenfield Investopedia. 2021. Proof of Work (PoW). [online] Available at:<https://www.investopedia.com/terms/p/proof-work.asp> [Accessed 10 July 2021]
[8] Laurence, T., n.d. Blockchain for Dummies, 2nd Edition. 2nd ed. John Wiley & Sons (US) © 2019.
[9] Garg, P., Garg, R., Prasad, R. and Mishra, A., 2015. A prospective study of ocular toxicity in patients receiving ethambutol as a part of directly observed treatment strategy therapy. *Lung India*, 32(1), p.16.
[10] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.
[11] X. Xu, I. Weber, M. Staples, L. Zhu, J. Bosch, L. Bass, C. Pautasso, and P. Rimba, "A taxonomy of blockchain-based systems for architecture design," in *2017 IEEE International Conference on Software Architecture(ICSA)*, April 2017, pp. 243–252.
[12] T. Sato and Y. Himura, "Smart-Contract Based System Operations for Permissioned Blockchain", 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS), Paris, pp. 1-6, 2018.
[13] Hyperledger Fabric, Available at:https://hyperledgerfabric.readthedocs.io/en/release-1.4/write_first_app.html
[14] Hyperledger Performance and Scale Working Group, "Hyperledger Blockchain Performance Metrics". Available at:
[15] https://www.hyperledger.org/wpcontent/uploads/2018/10/HL_Whitepaper_Metrics_P DF_V1.01.pdf.
[16] H. Sukhwani, J. M. Martínez, X. Chang, K. S. Trivedi and A. Rindos, "Performance Modeling of PBFT Consensus Process for Permissioned Blockchain Network (Hyperledger Fabric)", IEEE 36th Symposium on Reliable Distributed Systems (SRDS), Hong Kong, pp. 253-255, 2017.
[17] F. Benhamouda, S. Halevi and T. Halevi, "Supporting Private Data on Hyperledger Fabric with Secure Multiparty Computation", IEEE International Conference on Cloud Engineering (IC2E), Orlando, FL, pp.357-363, 2018.
[18] C. Cachin, "Architecture of the Hyperledger Blockchain Fabric", IBM Research - Zurich , 2016
[19] Hyperledger Architecture Working Group, "Hyperledger Architecture, Volume 1". Available at:https://www.hyperledger.org/wp-content/uploads/2017/08/Hyperledger_Arch_WG_Paper_1_Consensus.pdf.
[20] M. Valenta, P. Sandner, "Comparison of Ethereum, Hyperledger Fabric and Corda", Frankfurt School Blockchain Center, 2017