InterPlanetary File System Enabled Blockchain

Subham Sekhar Mohanty[1], Rounak Rathore[2], Revanth Reddy[3], K. Meenakshi[4]

[1,2,3]*Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, India*
[4]*Assistant Professor, Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, India*

---***---

Abstract - *Data preservation nowadays is being compared to be as precious as preservation of money. Data has never been so important, also neither this much in danger. Hackers and malicious software are always on their toes to steal the data and use it against our will and interests. Getting a bottle neck is never suggested. That's why we came up with a system which is decentralized as much as much possible. We are connecting Ethereum blockchain with Interplanetary File System (IPFS) instead of central server. Here the code, the storage and even the network is decentralized. Decentralized data gives everyone equal control and makes sure there are no bottlenecks. People always want their particulars, personals and essential date to be perfectly safe and secured and nobody should control and rule over except they, themselves. IPFS secures this feat. The IPFS is decentralized, peer to peer network and comes up with in a bundled with all the necessary credit that you get with bit torrent, Bitcoin and the web, and bind them up into one solid package. IPFS also comes up with deduplication and SHA-256 hashing algorithm for encrypting them securely. This is like all good things all together and nothing impure. This sounds perfect but we know that perfection is myth. IPFS has security and traceability concerns. That's why we are planned to introduce Ethereum blockchain. Blockchain being decentralized nodes and keeping track of everything. All the transactions of data.*

1. OBJECTIVE

To combine the highly decentralized IPFS with Ethereum blockchain using its smart contract technology with all the necessary terms mentioned in it and not just making it decentralized in all aspects but also no compromise with the security.

2. LITERATURE SURVEY:

2.1 Title: Smart contract programming languages on blockchains: An empirical evaluation of usability and security.

Description: The blockchain technology has been resulting in interventions in many industries for further developments and has tired good enough for many improvements and making lines easier in the world of blockchain, code is the law, whatever, once coded is irrevocable. The blockchain smart contract technology binds everything together and make all the transactions, secured, traceable and in irreversible. But the programming languages that are being used for implementing blockchain sometimes turnout to be incompetent. That's why this paper suggest to use a language which is native of blockchain and smart contracts technology. That language is solidity. Many efforts have been made regarding connecting it to some other OOP language such as Java, but couldn't satisfy the needs. Solidity is an object oriented language and uses the OOP concepts very efficiently to implement the smart contracts and naturally incorporating all the necessary components all together [22].

2.2 Title: A next generation smart contracts and decentralized application platform.

Description: The centralized storage is being replaced with distributed system digital contracts are coming into play with strong binding of codes and they can also be used for traceability of transactions [7]. The usage of Ethereum is enabling the programmers to genuinely descript the centralized flow of data and add peers to facilitate the data flow in chunks so that no malicious activity can destroy or disturb the data. Use of truffle framework can significantly make work easier and testing fruitful.

2.3 Title: IPFS – Content Addressed, Versionized, P2P file system.

Description: The IPFS can be looked upon as something to web in itself. It's like a single swarm of bit torrent and it exchanges object within one git repository. IPFS provides with a strong data structure over which all its data distribution is based upon and addressed data block-based encryption model. It's fully decentralized and accompanies with a lot of features helping us exempt ourselves a number of tedious jobs [1]. Using this peer-to-peer technology will help us to take decentralization to the ultimate level.

3. EXSITING SYSTEM

Current storage systems use a centralized server. If the central sever goes down everything goes down. Also every single piece of data being under one roof gives uncontrolled power to the storage companies. Also the governments also get the enormous tyrannical control over such servers and they can manipulate the data and keep an eye over it without the users' consent to facilitate their political agendas. For instance, in India, many times people try to whistle some information that common people should know, but the government tries to conceal that information for so called greater good. Government pressurizes these social media central storage servers to clear that data off the network. This way the government can exercise great deal of power unethically. Also IPFS all alone is not much of use although it's totally decentralized but once the owner uploads the file the hash file is shared with every one else that are on the peer network and the owner has no control over with which the other peers shares the hash file with so it has the problems of traceability, lack of privacy and violation of IP [7].

4. PROPOSED SYSTEM

In this paper, we are proposing a system with fully decentralized server and has come over all the limitations that IPFS (loose data and no traceability) is having which can rise a concern in the owners' heart. We are using blockchain. Using blockchain decentralizes almost everything. Using blockchain we are going to solve the problem of loose ended hash file which was being sent to every available peer [2]. Now this hash file will be stored at the blocks of blockchain. And the block chain as well as know is very secured using blockchain increases the effectiveness of decentralized by many folds.

5. SYSTEM ARCHITECTURE

5.1 The client
5.2 The IPFS
5.3 The block chain network

5.1 THE CLIENT

This is the person who actually owns the file and actually supposed to upload the file to the IPFS.

5.2 THE IPFS

It's the online decentralized network which received the file from the client and process it.

5.3 THE BLOCK CHAIN NETWORK

This network receives the hash file which is the key to access the file and stores it over its peers and its inaccessible without the key.

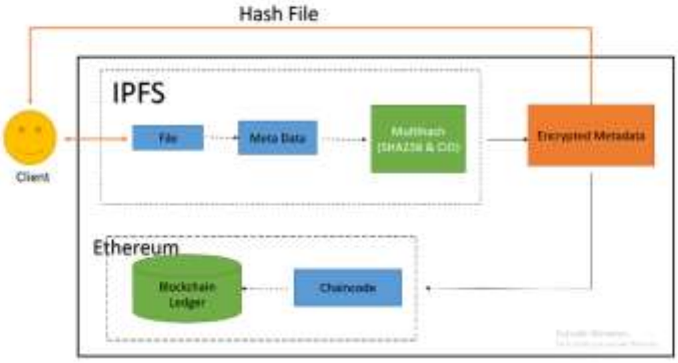


Fig 1: Architecture Diagram

When a file is uploaded by the client to the IPFS through the web app, it takes its meta data and encrypt that using SHA-256 and CID as a labelling instrument and creates a multihash key and stores the content of the file broken down in smaller chunks, shared over the p-to-p network. The key which is actually the encrypted metadata is shared with the user and stored over ethereum blockchain network totally secured.

6. PRELIMINARIES

- IPFS
- Etherium Blockchain
- Smart Contract
- NPM
- Truffle
- Ganache
- CID
- Web Browser
- Metamask

6.1 IPFS

The interplanetary file system can be accessed through its web application or software that can be installed on any browser. IPFS accompanies with necessary algorithms implemented in it. It used CID (Content Identifier) to get rid of duplicity in the content [17].

It is used SHA-256 and base-58 algorithms for encoding the data and it uses the market tree algorithm to retrieve the whole data when its provided with the hash key file [19][14].

6.2 ETHEREUM BLOCKCHAIN

The Ethereum is a permission less blockchain system which can be used freely as it provides the accounts with hundred fake Ethereum currency each. Its main feature is smart contract [9].

6.3 SMART CONTRACT

It's a computer protocol that is used to digitally facilitates the transactions and intended that both side of the parties keep their word and everything goes smoothly and securely this is written in solidity language. This is the native language for smart contracts [7][8].

6.4 NPM

Note package manager is a useful tool as it declivers necessary packages for the implementation of truffle framework [18].

6.5 TRUFFLE

Truffle frame work is used to write the smart contract for blockchain and for that purpose it creates facilitating dependencies.

6.6 GANACHE

It's the software that can be used to display the local Ethereum blockchain on your local host and incorporate it with the testing software.

6.7 CID

It's known as Content Identifier. It's a hashing and labelling technique that is used to encrypt the data over SHA-256 and then create a hash key file. It's also useful in eliminating duplicity [17].

6.8 WEB BROWSER

The implementation of ours is going to be in the form of an application. So a browser is necessary, although, google chrome is preferred specifically.

6.9 META MASK

This web application is to be used to communicate with our locally set up blockchain and constructs a connection between the web application and local blockchain setup.

7. MODULES

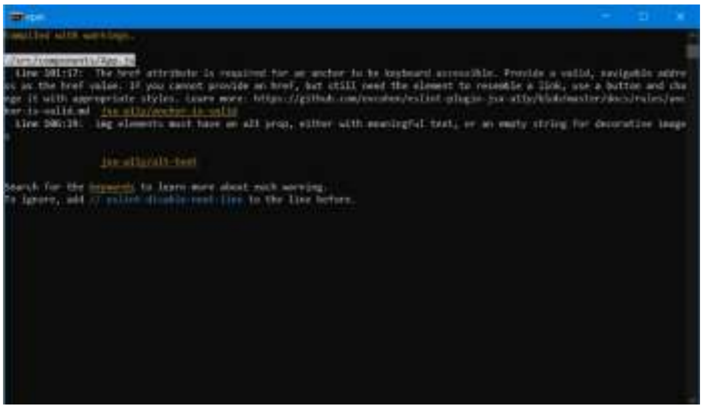
7.1 DEPLOYMENT OF THE SMART CONTRACT

The first step is the compiling of smart contracts and then their deployment. The directory is accessed using cmd, then we compile the smart contract that are stored in that particular directory using the commands: -

- Truffle migrate
- Truffle compile

Then the NPM is used to start the overall program using the command: -

- NPM run start

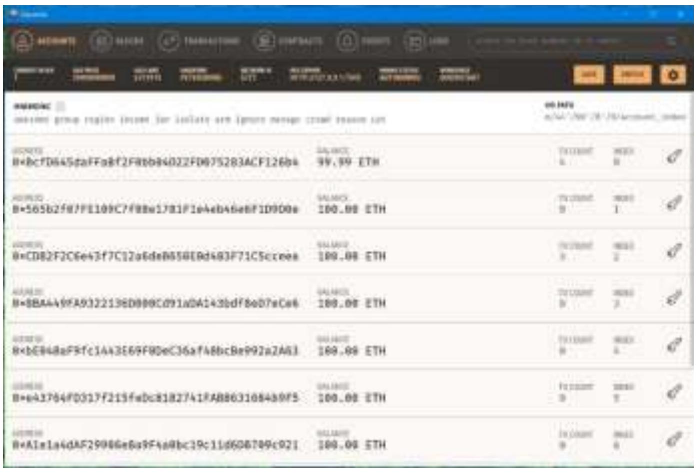


```
Compiled with warnings.

src/components/App.js
  Line 101:17:  The href attribute is required for an anchor to be keyboard accessible. Provide a valid, navigable addre
ss as the href value. If you cannot provide an href, but still need the element to resemble a link, use a button and cha
nge it with appropriate styles. Learn more: https://github.com/evcohen/eslint-plugin-jsx-a11y/blob/master/docs/rules/anc
hor-is-valid.md  jsx-a11y/anchor-is-valid
  Line 106:19:  img elements must have an alt prop, either with meaningful text, or an empty string for decorative image
s                jsx-a11y/alt-text

Search for the keywords to learn more about each warning.
To ignore, add // eslint-disable-next-line to the line before.
```

Fig 3(a): First module



```
ADDRESS                                      BALANCE     TX COUNT  INDEX
0xBcfD645daFFa8f2F0bb04022FD075283ACF126b4   99.99 ETH   4         0
ADDRESS                                      BALANCE     TX COUNT  INDEX
0x565b2f07FE109C7f08e1781F1e4eb46e6F1D9D0e   100.00 ETH  0         1
ADDRESS                                      BALANCE     TX COUNT  INDEX
0xCD82F2C6e43f7C12a6de8650E0d483F71C5cceea   100.00 ETH  0         2
ADDRESS                                      BALANCE     TX COUNT  INDEX
0x8BA449FA9322136D000Cd91a0A143bdf8e07eCe6   100.00 ETH  0         3
ADDRESS                                      BALANCE     TX COUNT  INDEX
0xbE848aF9fc1A43E69F0DeC36af48bcBe992a2A63   100.00 ETH  0         4
ADDRESS                                      BALANCE     TX COUNT  INDEX
0xe43764FD317f215feDc8182741FA88631084b9F5   100.00 ETH  0         5
ADDRESS                                      BALANCE     TX COUNT  INDEX
0xA1e1a4dAF29066e6a9F4a0bc19c11d6D8709c921   100.00 ETH  0         6
```

Fig 3(b): Setting up Ganache (Local Blockchain)

7.2 CONNECTING THE BLOCKCHAIN TO THE LOCAL HOST

The local host is connected to the block chain and to facilitate this process we mention the same local host address in our smart contracts which appears on the ganache software and has been used to deploy the local blockchain. The web application appears on the web browser. At the same time the meta mask is connected to the local block chain as well. Now the client can upload the file.

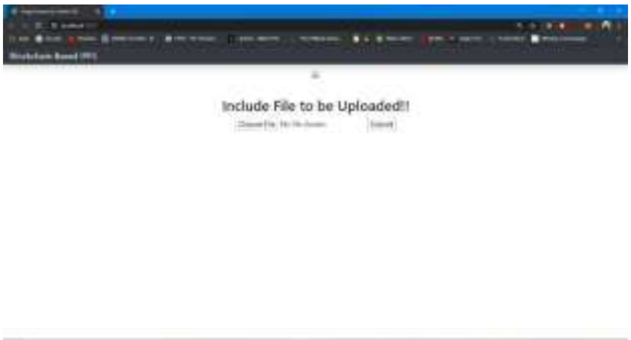


Fig 4: The User Interface of the Web Application

7.3 CONVERSION OF FILE

After the file has been uploaded the particular file is converted to a buffer and the buffer file is sent to the IPFS. This happens in the mean while when the currency transaction is pending. The buffer is displayed to us on the split screen.

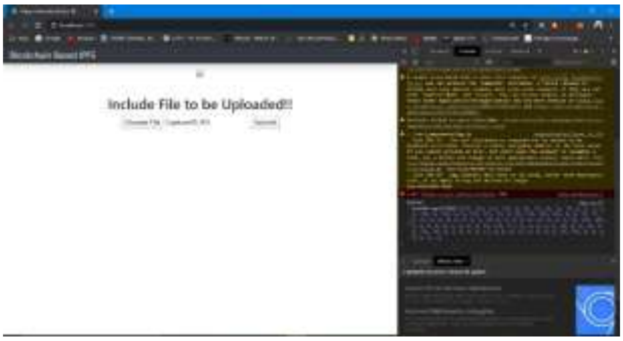
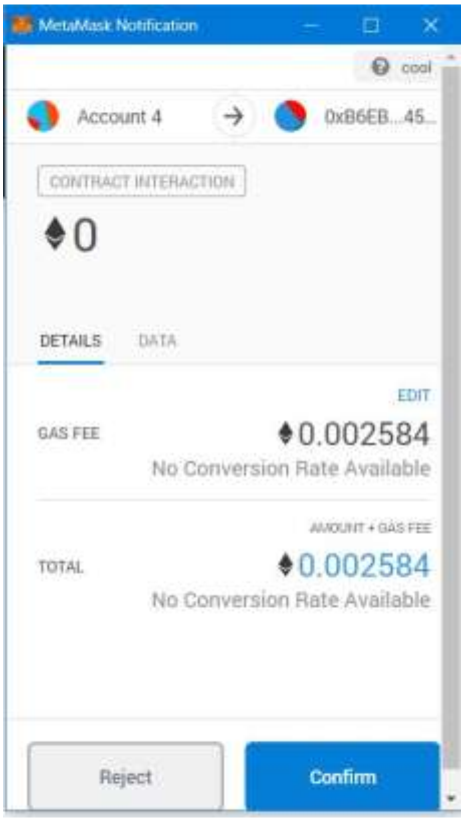


Fig 5: Conversion of File

7.4 TRANSACTION

When the user confirms the transaction from the meta mask pop up the IPFS goes further with the process at the back end. The IPFS encodes the uploaded file with SHA-256 and base 58 level of encoding and it creates a hash file which is shared with the client and is sent to different pairs on the local blockchain network [1][3][13]. Also the file is sent to different nodes at the IPFS network globally. The uploaded file can be viewed over the web application or the IPFS portal provided, the user has the hash key.


MetaMask Notification

cool

Account 4 → 0xB6EB...45...

CONTRACT INTERACTION

0

DETAILS DATA

EDIT

GAS FEE 0.002584
No Conversion Rate Available

AMOUNT + GAS FEE

TOTAL 0.002584
No Conversion Rate Available

Reject Confirm

Fig 6: The Ethereum Transaction shown on the Metamask

8. RESULTS AND DISCUSSION

We successfully established a connection between the locally set up Ethereum Blockchain and the IPFS distributed network and could share a file over to that network and were able retrieve it from some other node using the hash key shared with us. The transaction made within the Ethereum blockchain is helpful in tracing back the data and keeping it secured and find the culprit if its tempered with. Every piece of this arrangement is shared over the distributed network means no center and no chokepoint.

9. CONCLUSION

The IPFS equipped with the blockchain technology is perfect and tailor made solution to the centralized servers problem. All the transactions can be traced back with the help of block chain. The smart contract totally nullifies the requirement of the third party, hence, none controls our data in any way possible. It also saves us from the problem of duplicity.

This uses very high level of encryption algorithm which exempts us from worrying about the issue of counterfeiting. The IPFS with blockchain can be something which will be very useful and might be next level of storage for the future twenty years down the line.

10. REFERENCES

1] J. Benet, “IPFS-Content Addressed, Versioned, P2P File System”, arXiv:1407.3561v1, 2014.

[2] J. Yli-Huumo, D. Ko, S. Choi, S. Park, K. Smolander, “Where Is Current Research on Blockchain Technology?—A Systematic Review”, PLoS ONE 11(10): e0163477, 2016.

[3] Y. Chen, H. Li, K. Li and J. Zhang, “An improved P2P file system scheme based on IPFS and Blockchain”, IEEE International Conference on Big Data (Big Data), Boston, MA, pp. 2652-2657, 2017.

[4] Protocol Labs, “Filecoin: A Decentralized Storage Network”, 2017.

[5] 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.

[6] Hyperledger Performance and Scale Working Group, “Hyperledger Blockchain Performance Metrics”. Available at: https://www.hyperledger.org/wp-content/uploads/2018/10/HL_ Whitepaper _Metrics_PDF_V1.01.pdf.

[7] V. Buterin, “A Next Generation Smart Contract & Decentralized Application Platform”. Available at: http://blockchainlab.com/pdf/ Ethereum_white_paper-a_next_generation_smart_contract_and_ decentralized_application_platform-vitalik-buterin.pdf

[8] 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.

[9] M. Valenta, P. Sandner, “Comparison of Ethereum, Hyperledger Fabric and Corda”, Frankfurt School Blockchain Center, 2017.

[10] 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.

[11] C. Cachin, “Architecture of the Hyperledger Blockchain Fabric”, IBM Research - Zurich , 2016.

[12] 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.

[13] .L. Koonce, “The Wild, Distributed World: Get Ready for Radical Infrastructure Changes, from Blockchains to the Interplanetary File System to the Internet of Things”, Intellectual Property & Technology Law Journal , vol. 28, pp. 1-6, 2016.

[14] Base-58 Encoding, Available at: https://en.wikipedia.org/wiki/Base58

[15] IPFS Project, “IPNS”, Available at: https://docs.ipfs.io/guides/concepts/ipns/ [16] Hyperledger Fabric, Available at: https://hyperledgerfabric.readthedocs.io/en/ release-1.4/write_first_app.html

[17] CID. Available at: https://pascalprecht.github.io/posts/content-identifiers-in-ipfs

[18] Nodejs, Available at: https://nodejs.org/en/

[19] SHA-256, Available at: https://en.wikipedia.org/wiki/Secure_Hash_Algorithms

[20] JS-IPFS, Available at: https://docs.ipfs.io/introduction/install/

[21] Hyperledger Fabric, Available at: https://hyperledgerfabric.readthedocs.io/en/release-1.4/whatsnew.html

[22] R. M. Parizi, Amritraj, and A. Dehghantanha, “Smart Contract Programming Languages on Blockchains: An Empirical Evaluation of Usability