Journal of Information Technology and Digital World (2020)
Vol.02/ No. 01
Pages: 27-42
https://www.irojournals.com/itdw/
DOI: https://doi.org/10.36548/jitdw.2020.1.003

Fog Computıng – A Rasperry Pı Decentralızed Network

Dr. N. Bhalaji,
Department of Information Technology,
SSN College of Engineering, Kalavakkam,
Chennai, TamilNadu, India.
Email id:bhalajin@ssn.edu.in

Shanmuga Skandh Vinayak E
Department of Information Technology,
SSN College of Engineering, Kalavakkam,
Chennai, TamilNadu, India.
Email id:shanmugaskandhvinayak16095@it.ssn.edu.in

Abstract: Ever since the concept of parallel processing and remote computation became feasible, Cloud computing is at its highest peak in its popularity. Although cloud computing is effective and feasible in its usage, using the cloud for frequent operations may not be the be the most optimal solution. Hence the concept of FOG proves to be more optimal and efficient. In this paper, we propose a solution by improving the FOG computing concept of decentralization by implementing a secure distributed files system utilizing the IPFS and the Ethereum Blockchain technology. Our proposed system has proved to be efficient by successfully distributing the data in a Raspberry Pi network. The outcome of this work will assist FOG architects in implementing this system in their infrastructure and also prove to be effective for IoT developers in implementing a Raspberry Pi decentralized network while providing more security to the data.

Keywords: FOG Computing, Decentralized Application, data distribution, Blockchain, IPFS, Ethereum, smart contract, Truffle, Metamask, Raspberry Pi.

1.Introduction

In the wake of the global market starting to implement Cloud computing in the business model, the growth of the cloud computing market has gone from 58.6 billion US dollars in the year 2009 to 266.4 billion US dollars in year 2020 according to Statista [1]. It is estimated that this will reach an evaluation of 354.6 US billion dollars by the year 2022. In the result of such, the growth of FOG computing also tends to increase in its adaptability to the market implementing cloud architecture.

Unlike Cloud computing, FOG computing tends to provide data services that are less remote. The FOG stage of a network ballparks frequent and expeditious data in a network amongst an intermediate layer of peers between the cloud and the end edge devices. The basic structure of the FOG architecture is given in figure 1 [2].

Journal of Information Technology and Digital World (2020)
Vol.02/ No. 01
Pages: 27- 42
https://www.irojournals.com/itdw/
DOI: https://doi.org/10.36548/jitdw.2020.1.003

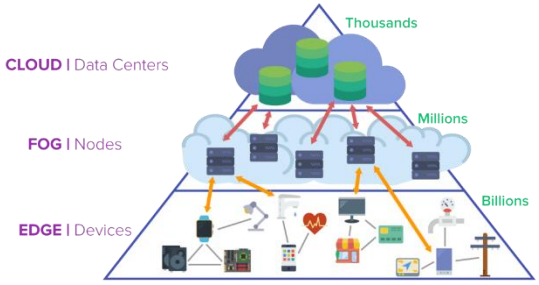

Fig. 1. FOG Architecture in Cloud Services

In this article, we propose a solution to improve the state of the FOG computing layer using the concept of decentralization and increasing security for data through blockchain technology. Blockchain is becoming one of the most popular technologies of the 2010s. According to Statista, the Blockchain users has grown from an approximate of 8 million to an approximate of 44 million users since 2016 to 2019 [3]. Figure 2. shows the growth of blockchain users form 2016 Q3.

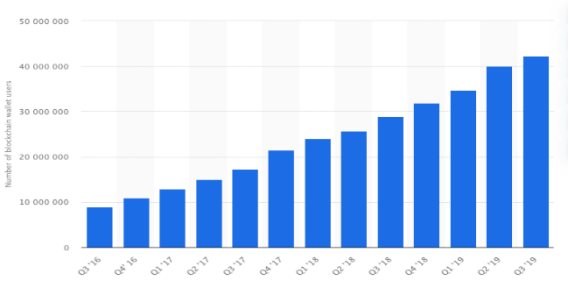

Fig. 2. Blockchain users from 2016 Q3

According to Statista, Blockchain is also being used in commercial solutions in both public and private sector [4]. Figure 3. shows the statistics of the blockchain impact on commercial sector.

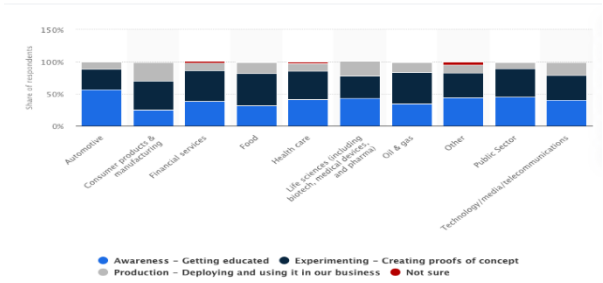

Fig. 3. Blockchain impact in Commercial Sector

These statistics show us that blockchain is being treated as a viable option to keep track of the

Journal of Information Technology and Digital World (2020)
Vol.02/ No. 01
Pages: 27- 42
https://www.irojournals.com/itdw/
DOI: https://doi.org/10.36548/jitdw.2020.1.003

transactions in a secure decentralized manner. Blockchains are being adopted as a trusted solution while being integrated in architectural framework of several industrial solutions.

The works of [9], the authors Bo Zhao, Peiru Fan and Mingtao Ni show the use of blockchain technologies along with the Infrastructure as a service to produce a system with high security and integrity in their cloud services to perform VM measurements. In the works of [10], the author Alexandru Stanciu implement an edge system utilizing the blockchain technology taking advantage of the Docker and Kubernetes containers to employ the edge nodes in the network. These works show the feasibility and the scope of blockchain technologies in the cloud environment to improve security, stability and integrity. The authors Zehui Xiong, Yang Zhang, et al., in this work [11] implement a mobile edge computing network utilizing the blockchain technology to improve security by Proof – of – Work for the sensors data in their network such as accelerometer, GPS, etc. This work gives an aspect of the feasibility of blockchain technology in the IoT environment and an example of its application. Similarly, in the scope of utilizing the FOG computing concept to serve the blockchain technology, the authors Mayra Samaniego, Uurtsaikh Jamsrandorj and Ralph Deters of University of Saskatchewan has proposed a FOG based network hosting system in their work [12]. This work further improves the scope of moving from an edge based blockchain network to a pure FOG based system to host applications.

In this article, a raspberry pi based decentralized network that hosts a decentralized application capable of distributing user data in the network is implemented and the results are studied.

2.Related works

Quanqing Xu, Zhiwen Song, et al., [13] have implemented a social network using the concept of decentralized networking and decentralized application. This is achieved using the following technologies, IPFS – The Inter Planetary File System, Ethereum Blockchain, Solidity for creating the smart contract and ReactJS for the front end features of the application. The application posts the content .i.e. tweets using the website and distributing the data throughout the network and securing the data information onto a blockchain. A peer – to – peer network is implemented with different accounts on the network to post tweets, which included text and images.

In the paper [14], the authors N. Nizamuddin, K. Salah, et al., demonstrate the implementation of a blockchain based document version control and tracking mechanism using the concept of decentralized validation. When a new version of an existing documents or any changes are made to the existing documents, the file is published in the Interplanetary File System and the changes are sent to the validating group. The versions and the changes made to a document is made to be decentralized and available to the public peers which can be validated by the peers on the network or a customized group of peers to publish the changes in the network, that would keep in track of all of the validations.

In the basis of the working of the IPFS that allows the users to keep a copy of their data with themselves

Journal of Information Technology and Digital World (2020)
Vol.02/ No. 01
Pages: 27- 42
https://www.irojournals.com/itdw/
DOI: https://doi.org/10.36548/jitdw.2020.1.003

and publish the same in the network, a similar work is done by the authors M. Alessi, A.Camillò, et al., in their work[15]. The authors have implemented a decentralized data store for the safekeeping of sensitive information of the user while integrating such a system with the users' social media. This is in accordance to the fact that social media tend to use our personal data. The proposed system called as the Personal Data Store (PDS), is able to provide the user with an option choosing the data that can be used by the Servify ecosystem generated based on their interest using their social media accounts.

The authors Shangping Wang, Yinglong Zhang and Yaling Zhang provided a security-based solution for a decentralized application in their work [16]. The authors utilize the traditional IPFS technology to distribute files in multiple nodes. But along with this the system provides an additional level of security to the data by providing private keys between the sender and the recipient. When the data is published, the user desiring to use the data requests for a key to access the file, for which the data owner provides the user with an AES encrypted key in a private channel. Once the user obtains the data, the owner provides another key obtained from a keyword in the file to the user in a private channel and the user decrypts the file. Furthermore the system also provides a blockchain level based transaction book keeping system to track the owner ship of the data in the network.

The authors Pengfei Wang, Wenjuan Cui, and Jianhui Li have developed a complete decentralized application that is capable of storing and distributing data in the IPFS network and providing security to that data through the use of blockchain network, called the uPort in their work [17]. The application consist of a UI that allows the user to login to their Ethereum account and view their address using the application. This is also used to view the user's public address and the ether balance. The UI consist of a page to view and upload the files into the IPFS network. The user uses a certain amount of gas while uploading a file and similarly while obtaining the file through the blockchain to execute the smart-contract. The authors also implement a security mechanism to authenticate and verify the user's Ethereum account and the validity of any transaction using the uPort ID system. In the scope of implementing a fully-fledged application for a commercial-ready use, the author Jay Nikhil and Bibodi [18], implemented a complete decentralized application to use and share podcast audio files with all the members of the network called the Podweb. The application consists of a UI, that allows the user to sign up and login to the system. The system uses Stripe to carry out payments, allowing the users to buy more ethers from the initial ethers awarded at sign up to purchase podcasts. The user is then allowed to upload their podcasts either for free or provide it with a cost to buy. Once the user publishes the podcast the file is distributed in the IPFS network and the reference of the file is stored in the Ethereum blockchain. This is then notified to all the participants on the network and the users can either pay in Ether, the cost of the podcast to be able to own them or play the file for free based on the owner's purchase policy. Similarly, the authors Bokang Jia, Chenhao Xu, et al., in their work [19] have implemented a decentralized application with the purpose of sharing music files in a distributed manner. The application named Opus, is a backend system that is capable of obtaining music from the users, i.e. artists and distributing them in the network. The files are distributed on a peer-to-peer networks such as the IPFS and the security curation is done by the blockchain technology. This backend system provides an API service to the UI that allows artists to upload music files into the system and to specify the ether amount for the respective music to be used by the end user that alerts the users of new activities

Journal of Information Technology and Digital World (2020)
Vol.02/ No. 01
Pages: 27- 42
https://www.irojournals.com/itdw/
DOI: https://doi.org/10.36548/jitdw.2020.1.003

on the network.

In the article[20], the system is based on IoT devices in the scope of extending the applications of the devices into the IoT environment. The authors Bin Liu, Xiao Liang Yu, et al., have implemented such a framework in their work, that is capable of securely storing the IoT data of a network in a distributed and secure manner. This system deals with the continuous real time data that is the IoT devices generate and distribute them in the network and store it in the cloud for long term use. This system is capable of maintaining the integrity of the IoT data and storing them securely. The authors Kumar, T. S et al [36] presents the better allocation management in FOG for QOS improvement, Kumar, R et al [37] perfromed the "A novel report on architecture, protocols and applications in Internet of Things (IoT)." Bestak, R et al [38] conducted the “Big Data Analytics for Smart Cloud-Fog Based Applications” Raj, Jennifer S et al [39] put forth the. "A dynamic overlay approach for mobility maintenance in personal communication networks." Ananthi, J. V et al [40] proffered the “Automation Using Iot in Greenhouse Environment”

3.Techniques and tools

The experiment consists of two important tools and one hardware component to attain distribution and security. The Interplanetary File System and the Ethereum Blockchain technology are the tools used and the component used is the Raspberry Pi. The Interplanetary File System is an open-source tool that is capable of attaining distribution throughout the network and Ethereum is an opensource blockchain platform capable of keeping track of the transactions. The Raspberry Pi computers functions as peers on the network and are used to test the usability of the FOG technique using these tools and hosting the decentralized application.

3.1.The Interplanetary File System:

The Interplanetary File System [5] is an open-source file system that distributes the data from one peer to all the peers on the network. The IPFS follows a content-based protocol as opposed to the conventional location-based protocol followed by the HTTP/HTTPS standards. When the data is allowed to be stored in the IPFS, the file system splits the data in portions of 256 kilobytes blocks of data. These portions are then hashed using SHA 256 algorithm along with base58 hashing and distributed in the network. And a final hash pointing these hashes is created that is used to locate the data in the network. The Interplanetary File system works similar to a Torrent network to locate files using content-based technique. The peers in the network is able to communicate to one another in search for the content using the hash that they require by searching the nearest peer. This search algorithm is made efficient by the IPFS through the use of Merkle DAG algorithm.

Journal of Information Technology and Digital World (2020)
Vol.02/ No. 01
Pages: 27- 42
https://www.irojournals.com/itdw/
DOI: https://doi.org/10.36548/jitdw.2020.1.003

3.1.1.Merkle Tree Algorithm

A Merkle Tree is a data structure to store cryptographic hashes in terms of related blocks that point to one another in a directed manner to a singular hash or the root hash. This algorithm is used in keeping track of the hashes that are generated from the data split in the IPFS distribution mechanism. Since the Merkle tree resembles a binary tree, the search time to find the hash in the network will be of the form O(n). The hashes generated at each node is of the form (1)

$$\text{hash node} = \text{hash}(\ \text{hash(child 1)} + \text{hash(child 2)}\ ). \quad (1)$$

The structure of the Merkle tree is given in figure 4.

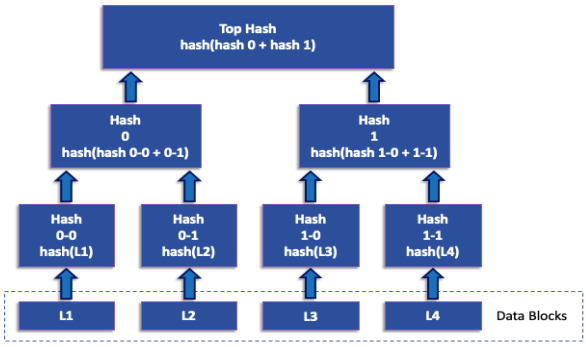

Fig. 4. Merkle Tree

3.1.2.Directed Acyclic Graph

The Directed Acyclic Graph is a data structure that represents the flow of control in a process in terms of nodes and edges that do not form any cycles. This graph is used here in IPFS distribution to utilize its algorithm of infinite edges and vertices by allowing peers to communicate with each other in such a manner during distribution. This mechanism is also used during the identification of the peers in the network while discovering the data in the peers.

Since the DAG algorithm has the ability to provide an infinite number of edges, it is possible to find the shortest path from any peer to another that has the data available. A structure of the DAG is given in figure 5.

Journal of Information Technology and Digital World (2020)
Vol.02/ No. 01
Pages: 27- 42
https://www.irojournals.com/itdw/
DOI: https://doi.org/10.36548/jitdw.2020.1.003

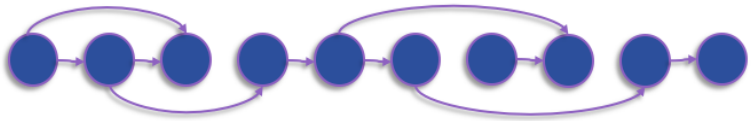

Fig. 5. A Directed Acyclic Graph

In a DAG, the vertices should be of the form $x \leq y \leq z$, where $\leq$ denotes partial order. The node z can be reached by both x and y, but the node y can be reached by only x and no other node can reach x.

3.1.3.Merkle DAG

The combination of the concept of Merkle tree and a DAG is used in the IPFS mechanism to keep in track of the hashed data along with the path to the data discovery. When the Merkle tree is formed, the DAG updates itself on the possible internal paths to the data to provide the shortest path during the data retrieval. The advantage of a Merkle DAG is that the Merkle tree need not be balanced on either side of the root node, therefore by creating shortest path along with the rooted hash tree. Using these two mechanisms, the efficiency of the protocol is increased.

3.2.Ethereum Blockchain

The Ethereum tool is a blockchain technology that is implemented in this experiment for the purpose of enhancing the security and to improve the usability of the IPFS technology. A blockchain is a data structure that records data, transactions in this scenario, in a well-defined double linked data structure. The Ethereum is an open source blockchain component that allows developers to utilize its consensus protocols to implement a secure network. Similar to the Java Virtual Machine, Ethereum is deployed and run on an Ethereum Virtual Machine on which the Ethereum network deploys and executes the smart contracts [3.2.1] during transactions. The Ethereum blockchain follows the Proof-of-Work consensus protocol [3.2.2] while validating the transactions. The applications using the blockchain technology utilizes an incentive called Ether, native to the Ethereum platform in order to initiate and validate the transactions, which serves as a pseudo currency to mine the transactions.

3.2.1.Smart Contract

Similar to a real-life contract, a smart contract is an agreement between the vendors and the peers using the services of the vendors by following the rules of the contract. The contract functions as a mechanism for the peers to execute the rules when a certain criterion is met. But unlike a real-life contract, a smart contract is capable of executing itself on the Ethereum platform without any intervention by the peers utilizing the services. When any new blockchain is created, a smart contract is deployed at the beginning such that any block created in the blockchain should have followed and utilized the contract before

Journal of Information Technology and Digital World (2020)
Vol.02/ No. 01
Pages: 27- 42
https://www.irojournals.com/itdw/
DOI: https://doi.org/10.36548/jitdw.2020.1.003

being added to the chain.

In our experiment, the smart contract is shown in figure 6.

```
pragma solidity ^0.5.0;

contract Store {
  string ipfs_hash;

  function set (string memory hash) public {
    ipfs_hash = hash;
  }

  function get () public view returns (string memory) {
    return ipfs_hash;
  }
}
```

Fig. 6. Smart contract used in experiment

The above smart contract is used to insert the IPFS hash into the blockchain and retrieve the same. The set () function sets the IPFS hash in the local memory for the current session and to the current block being mined. The get () function retrieves the hash from the latest block.

3.2.2.Proof-of-Work consensus protocol

In the blockchain environment, a block is validated before it is added to the blockchain. This consensus protocol works such that, at least more than half of the peers on the network agree to the validity of the block. This functions in a manner that, if the majority of the vote on the block is agreed upon to be valid, then the block is considered valid. The Proof-of-Work is to award the peer based on the computational resource provided in validating the block.

3.3.Raspberry Pi

The Raspberry Pi is a single board mini computer designed and created by the Raspberry Pi Foundation in the United Kingdom. This computer is capable of performing basic tasks with minimum configuration that a fully-fledged computer is capable of performing. Since the FOG concept can be implemented for the Internet of Things architecture as well, this specific component was chosen in order to test its feasibility of running a private, fully decentralized application. The Raspberry Pi used in this experiment is the Raspberry Pi version 3 B Plus, having a Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC clocking at 1.4GHz overclocked to 1.54GHz, 1GB of LPDDR2 SDRAM and 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, making it suitable for the component to perform as a server for a small numbered cluster. The Raspberry Pi is operated using Raspbian, a Linux based operating system tailored for the Raspberry Pi. The Raspberry Pi is given in the figure 7.

Journal of Information Technology and Digital World (2020)
Vol.02/ No. 01
Pages: 27- 42
https://www.irojournals.com/itdw/
DOI: https://doi.org/10.36548/jitdw.2020.1.003

Fig. 7. Raspberry Pi 3 B Plus

4.Experimental Setup and Results

The experimental setup consists of 4 Raspberry Pi 3 B Plus operated using the Raspbian Operating system. The IPFS daemon instance is running on all the peers along with the configured bootstrap file for the IPFS to connect all the nodes on the network. The IPFS is made to maintain this state and listen to a dedicated port for all the incoming traffic. The EVM is made active by realizing the Geth [6] package utilizing the Go Language by Google Inc. The Instance is initiated through a custom Genesis file for this instance of the blockchain. The Genesis file is a set of rules and configurations that the blockchain will follow during its creation and execution. The genesis file is given in the figure 8.

```
{
   "config": {
      "chainId": 1969,
      "homesteadBlock": 0,
      "eip155Block": 0,
      "eip158Block": 0,
      "byzantiumBlock": 0
   },
   "difficulty": "1",
   "gasLimit": "2000000",
   "alloc": {
      "6ace66e0f69678358725654efc67e672f25d4dc4": {
         "balance": "10000000000000000000000"
      },
      "ae13d41d66af28380c7af6d825ab557eb271ffff": {
         "balance": "12000000000000000000000"
      }
   }
}
```

Fig. 8. Genesis File data

After instantiating the Genesis Block by executing the Genesis file, a new blockchain instance is created along with all the peers being connected to this instance by providing a static-nodes.json file in the Geth directory. Considering the fact that the mining time increases gradually during the mining process of blocks in the blockchain, the consensus time must be kept at constant, to avoid memory overload on the Raspberry Pi. This is achieved by editing the consensus file for the installation in each Raspberry Pi and setting the time (in milliseconds) to a low value. Figure 9 shows the consensus file for an instance of a Raspberry Pi.

```
func CalcDifficulty(config *ChainConfig, time, parentTime uint64, parentNumber, parentDiff *big.Int) *big.Int {
    return big.NewInt(1)
}
```

Fig. 9. Consensus.go file of a Raspberry Pi

The Geth instance is started as a daemon process with the auto DAG generation disabled to avoid the

Journal of Information Technology and Digital World (2020)
Vol.02/ No. 01
Pages: 27- 42
https://www.irojournals.com/itdw/
DOI: https://doi.org/10.36548/jitdw.2020.1.003

Raspberry Pi running out of memory. The JavaScript console, native to the Ethereum platform is used to find the details on the current instance of the blockchain. The peers connected to the current network can be found using the command admin.peers in the JavaScript console. The peer information of the one of the peers is given in figure 10.

Fig. 10. Peer information of the network

The total number of peers connected to a particular peer can be found using the net.peerCount command in the JavaScript console. Figure 11 shows the JavaScript console output for the net.peerCount command.

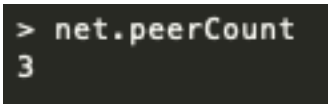

Fig. 11. JavaScript console peer count of the network

When a new transaction is initiated, the Ethereum network requires the peer to provide a certain amount of incentive to validate the transaction. This incentive is called Gas. It is measured in Gewi. Gas is the measure on the amount of computational power required to process a block, whereas Ether is the standard currency measure in the Ethereum platform.

$$1 \text{ Gwei} = 10^{-9} \text{ Ether} \quad (2)$$

This Gwei is the standard used in the Genesis File [Fig. 4]. A fraction of this Ether is used depending on the Gas price set by the Genesis File and is utilized for every transaction throughout the network whenever the smart contract is executed to create a block. The ether balance* for any peers' account can be found in the JavaScript console function eth.getBalance(eth.coinbase). Figure 12 shows the JavaScript console output for the eth.getBalance(eth.coinbase) command.

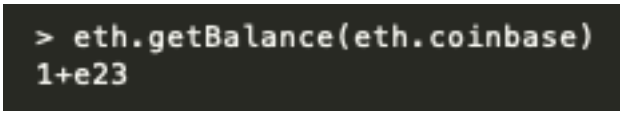

Fig. 12. JavaScript console showing ether balance for the account

Journal of Information Technology and Digital World (2020)
Vol.02/ No. 01
Pages: 27- 42
https://www.irojournals.com/itdw/
DOI: https://doi.org/10.36548/jitdw.2020.1.003

*These ethers do not represent the real ether and can only be transferred to any account on the private network.

The Ethereum platform is instantiated with the smart contract using the Truffle Blockchain Framework [7] after the creation and initiation of the Genesis block. This process is done only once and the blockchain follows the current smart contract throughout the execution cycle. Once the smart contract is deployed, Metamask [8], an Ethereum bridge tool that is capable of interacting with the Ethereum IPC instance is utilized to initiate transactions. This process is given in the figure 13.

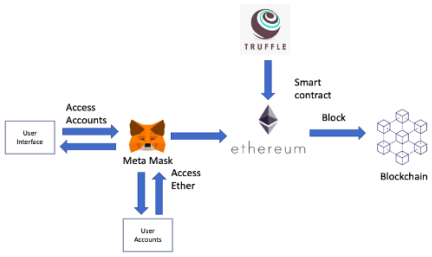

Fig. 13. Smart Contract initiation using Truffle

The User Interface files and the smart contract are published in the IPFS, such that the UI elements are not centralized either. The back-end algorithm of the IPFS and the Ethereum instance is given in table.1

Table. 1. Pseudo code for Back-End Algorithm

Pseudo code for Back-End Algorithm

```
Back-End Algorithm
Begin
        Instantiate IPFS and Ethereum with smart contract.
        Repeat
        if (Request==POST file)
            Get File
            Add to IPFS
            Obtain IPFS hash
            if (IPFS hash)
                Deploy smart contract
                Create Transaction
                Generate Block
                Mine Block
                if (mine==successful)
                    Add to block to blockchain
        if (Request==GET file)
            Read hash from blockchain using smart contract
            if(hash)
                Get hash content from IPFS
        End
```

Journal of Information Technology and Digital World (2020)
Vol.02/ No. 01
Pages: 27- 42
https://www.irojournals.com/itdw/
DOI: https://doi.org/10.36548/jitdw.2020.1.003

End

The user is prompted to upload the files in the index page. Once the file is uploaded, the Metamask UI (Figure 14) prompts the user to pay for the transaction. After the block is mined, the files appear on the initial index page (Figure 15).

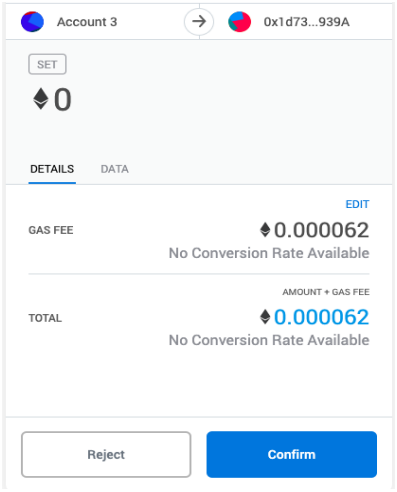

Fig. 14. Metamask Transaction Prompt

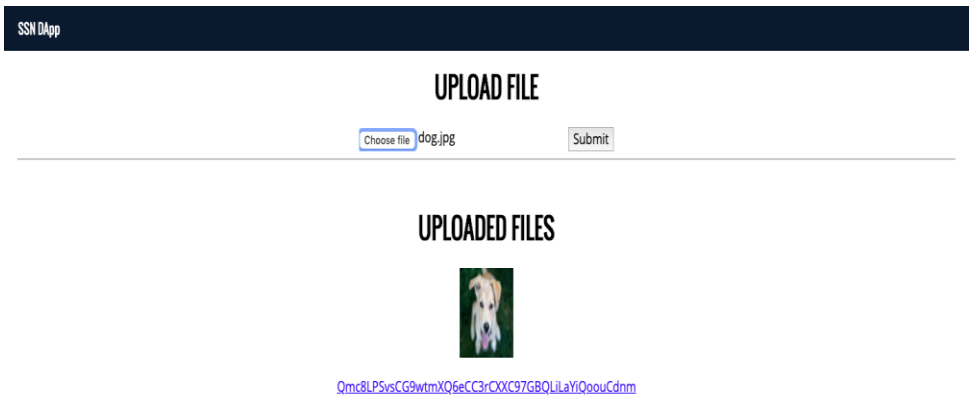

Fig. 15. Decentralized App Index Page

The IPFS hash of the files are stored in the blockchain that are decentralized and distributed throughout the network to all the peers. The blocks are added to the chain in the order that they are created. The IPFS hashes are obtained from the blockchain using the smart contract. Hence any peer trying to tamper with the blockchain must alter at least 51 percent of the peers to cause liable alteration in security, which is exponentially increased in difficulty as the size of the network peer count increases. The applications utilizing the blockchain can now access the blockchain as a single point of entry to access files, thereby decreasing search load and increasing security. The sequence of operations is given in figure 16.

Journal of Information Technology and Digital World (2020)
Vol.02/ No. 01
Pages: 27- 42
https://www.irojournals.com/itdw/
DOI: https://doi.org/10.36548/jitdw.2020.1.003

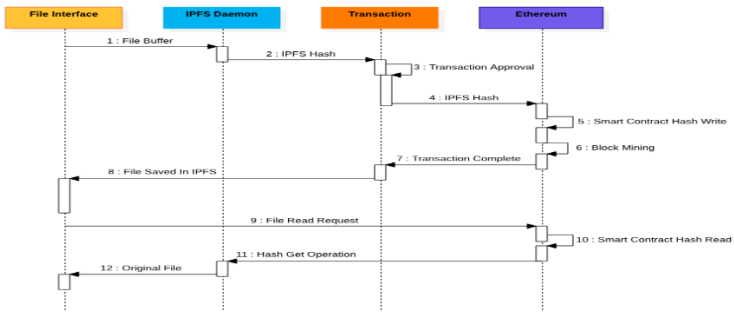

Fig. 16. DApp Operations Sequence

During the validation of the blocks for a transaction, the computational time and the computational difficulty decreases with an increase in the number of peers mining the network. Since the difficulty to mine the block remains constant for the entire network, the mining pool increases the computational utilization. Table 2. shows the CPU usage and the time taken to mine a block by the peers.

Table 2. Mining CPU usage and Time taken per Block

No. of Peers (Raspberry Pi)	CPU Usage % (Geth)	Avg. Time/Block (seconds)
2	12	9
3	7	5
4	4.2	3

These mining operations can be improved significantly by introducing ASIC machines into the network and an adaptive difficulty can be implemented for the same to improve security.

5.Conclusion and Future Works

In this paper we have demonstrated the deployment of a file decentralized application, deployed on a network consisting of Raspberry Pi. This experiment provides an insight on

- The feasibility of a Raspberry Pi for a small cluster decentralized network.
- The usability of the network and the components to deploy smart contracts and maintaining a blockchain.
- The adequacy of the network constraints to deploy and maintain a decentralized application.

Considering the facts that these are the features exhibited by a FOG network, this system can be deployed for a small-scale FOG computing and file sharing architecture during the integration of this system with the cloud architecture.

Journal of Information Technology and Digital World (2020)
Vol.02/ No. 01
Pages: 27- 42
https://www.irojournals.com/itdw/
DOI: https://doi.org/10.36548/jitdw.2020.1.003

Since this system is made up of Raspberry Pi, a popular component choice amongst the Internet-of-Things developers, the system can also be used to deploy in an IoT architecture. Thus, rendering the IoT system in such manner will acheive a tamper-proof network while fabricating an efficient network, utilizing the decentralized data for high data availability.

References

[1] https://www.statista.com/statistics/273818/global-revenue-generated-with-cloud-computing-since-2009/
[2] https://erpinnews.com/fog-computing-vs-edge-computing
[3] https://www.statista.com/statistics/647374/worldwide-blockchain-wallet-users/
[4] https://www.statista.com/statistics/878748/worldwide-production-phase-blockchain-technology-industry/
[5] https://ipfs.io
[6] https://geth.ethereum.org
[7] https://www.trufflesuite.com
[8] https://metamask.io
[9] Bo Zhao, Peiru Fan and Mintao Ni, 2018 Mchain: A Blockchain-based VM Measurements Secure Storage Approach in IaaS Cloud with Enhanced Integrity and Controllability. (2018). IEEE Access, 1–1.doi:10.1109/access.2018.2861944.
[10] Stanciu, A. (2017). Blockchain Based Distributed Control System for Edge Computing. 2017 21st International Conference on Control Systems and Computer Science (CSCS).doi:10.1109/cscs.2017.102.
[11] Xiong, Z., Zhang, Y., Niyato, D., Wang, P., & Han, Z. (2018). When Mobile Blockchain Meets Edge Computing. IEEE Communications Magazine, 56(8), 33–39.doi:10.1109/mcom.2018.1701095.
[12] Samaniego, M., & Deters, R. (2016). Blockchain as a Service for IoT. 2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData). doi:10.1109/ithings-greencom-cpscom-smartdata.2016.102.
[13] Xu, Q., Song, Z., Mong Goh, R. S., & Li, Y. (2018). Building an Ethereum and IPFS-Based Decentralized Social Network System. 2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS).doi:10.1109/padsw.2018.8645058.
[14] Nizamuddin, N., Salah, K., Ajmal Azad, M., Arshad, J., & Rehman, M. H. (2019). Decentralized document version control using ethereum blockchain and IPFS. Computers & Electrical Engineering, 76, 183–197.doi:10.1016/j.compeleceng.2019.03.014.
[15] M. Alessi, A. Camillo, E. Giangreco, M. Matera, S. Pino and D. Storelli, "Make Users Own Their Data: A Decentralized Personal Data Store Prototype Based on Ethereum and

IPFS," 2018 3rd International Conference on Smart and Sustainable Technologies (SpliTech), Split, 2018, pp. 1-7.

[16] S. Wang, Y. Zhang and Y. Zhang, "A Blockchain-Based Framework for Data Sharing With Fine-Grained Access Control in Decentralized Storage Systems," in IEEE Access, vol. 6, pp. 38437-38450, 2018. doi: 10.1109/ACCESS.2018.2851611.

[17] Wang P., Cui W., Li J. (2019) A Framework of Data Sharing System with Decentralized Network. In: Li J., Meng X., Zhang Y., Cui W., Du Z. (eds) Big Scientific Data Management. BigSDM 2018. Lecture Notes in Computer Science, vol 11473. Springer, Cham.

[18] Bibodi and Jay Nikhil 2008 PodWeb : a decentralized application powered by Ethereum network. Sacramento Masters Projects, URI: http://hdl.handle.net/10211.3/207993.

[19] Bokang Jia, Chenhao Xu, Rehan Gotla, et al., 2016 Opus - Decentralized music distribution using InterPlanetary File Systems (IPFS) on the Ethereum blockchain V0.8.3.

[20] B. Liu, X. L. Yu, S. Chen, X. Xu and L. Zhu, "Blockchain Based Data Integrity Service Framework for IoT Data," 2017 IEEE International Conference on Web Services (ICWS), Honolulu, HI, 2017, pp.468-475. doi: 10.1109/ICWS.2017.54.

[21] Sivakumar P and Dr. Kunwar Singh, Privacy based decentralized Public Key Infrastructure (PKI) implementation using Smart contract in Blockchain. National Institute of Technology, Trichy, Tamil Nadu 620015.

[22] W. Cai, Z. Wang, J. B. Ernst, Z. Hong, C. Feng and V. C. M. Leung, "Decentralized Applications: The Blockchain-Empowered Software System," in IEEE Access, vol. 6, pp. 53019-53033, 2018. doi: 10.1109/ACCESS.2018.2870644.

[23] Debajani Mohanty, Ethereum for Architects and Developers with Case Studies and Code Samples in Solidity. Chapter 6.

[24] Keyur Paralkar, Shiwani Yadav, Shikha Kumari, et al., 2018 PHOTOGROUP: DECENTRALIZED WEB APPLICATION USING ETHEREUM BLOCKCHAIN. International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 05 Issue: 04 | Apr-2018.

[25] Huang, H., Li, K.-C., & Chen, X. (2018). Blockchain-based fair three-party contract signing protocol for fog computing. Concurrency and Computation: Practice and Experience, e4469. doi:10.1002/cpe.4469.

[26] Khan, M. A., & Salah, K. (2018). IoT security: Review, blockchain solutions, and open challenges. Future Generation Computer Systems, 82, 395–411. doi:10.1016/j.future.2017.11.022.

[27] Y. Li, Z. Zhang, Y. Peng, H. Yin, and Q. Xu, "Matching user accounts based on user generated content across social networks," Future Gen- eration Computer Systems, vol. 83, pp. 104-115, 2018.

[28] Xu Q., Aung K.M.M., Zhu Y., Yong K.L. (2018) A Blockchain-Based Storage System for Data Analytics in the Internet of Things. In: Yager R., Pascual Espada J. (eds) New Advances in the Internet of Things. Studies in Computational Intelligence, vol 715. Springer, Cham.

[29] Hardjono, T., & Smith, N. (2016). Cloud-Based Commissioning of Constrained Devices using Permissioned Blockchains. Proceedings of the 2nd ACM International Workshop on IoT Privacy, Trust, and Security - IoTPTS '16.doi:10.1145/2899007.2899012.

[30] Confais, B., Lebre, A., & Parrein, B. (2017). An Object Store Service for a Fog/Edge Computing Infrastructure Based on IPFS and a Scale-Out NAS. 2017 IEEE 1st International Conference on Fog and Edge Computing (ICFEC).doi:10.1109/icfec.2017.13.

[31] Ali, M. S., Dolui, K., & Antonelli, F. (2017). IoT data privacy via blockchains and IPFS. Proceedings of the Seventh International Conference on the Internet of Things - IoT '17. doi:10.1145/3131542.3131563.

[32] Antonio Tenorio-Fornés, Viktor Jacynycz, David Llop, et al., 2019 Towards a Decentralized Process for Scientific Publication and Peer Review using Blockchain and IPFS. Proceedings of the 52nd Hawaii International Conference on System Sciences 2019.

[33] Marco Conoscenti, Antonio Vetro and Juan C. D. Martin. 2016 Blockchain for the Internet of Things: A Systematic Literature Review. In Proceeding of The Third International Symposium on Internet of Things: Systems, Management and Security (IOTSMS-2016).

[34] M. A. Rahman, M. M. Rashid, M. S. Hossain, E. Hassanain, M. F. Alhamid and M. Guizani, "Blockchain and IoT-Based Cognitive Edge Framework for Sharing Economy Services in a Smart City," in IEEE Access, vol. 7, pp. 18611-18621, 2019. doi: 10.1109/ACCESS.2019.2896065.

[35] Siraj Raval, Decentralized Applications HARNESSING BITCOIN'S BLOCKCHAIN TECHNOLOGY.

[36] Kumar, T. S. (2019). Efficient Resource Allocation and QOS Enhancements of Iot with Fog Network. Journal of ISMAC, 1(02). 101-110.

[37] Kumar, R. Praveen, and S. Smys. "A novel report on architecture, protocols and applications in Internet of Things (IoT)." In 2018 2nd International Conference on Inventive Systems and Control (ICISC), pp. 1156-1161. IEEE, 2018.

[38] Bestak, R., & Smys, S. (2019). Big Data Analytics for Smart Cloud-Fog Based Applications. Journal of trends in Computer Science and Smart technology (TCSST), 1(02), 74-83.

[39] Raj, Jennifer S., and R. Harikumar. "A dynamic overlay approach for mobility maintenance in personal communication networks." Peer-to-Peer Networking and Applications 7, no. 2 (2014): 118-128

[40] Raj, J. S., & Ananthi, J. V. (2019). Automation Using Iot in Greenhouse Environment. Journal of Information Technology, 1(01), 38-47.