Decentralized File Transfer System Blockchain-based File Transfer

Anusree K, Jagan Sathiaseelan Vadekkat, Abhinu R Dev, Abhinav
Department of Computer Science & Engineering
Sahrdaya College of Engineering and Technology,
Thrissur, Kerala, India

***Abstract*—Many peer-to-peer file-sharing programs have been developed around the world. Some applications have entirely failed, while others are still alive and well. BitTorrent is the latter, and it is capable of organizing networks of untrustworthy peers (swarms) to cooperate in the distribution of file fragments to one another. When compared to HTTP, however, these applications cannot address all use cases. HTTP, as we all know, is one of the most widely used file distribution systems in the world. Because of the rapid growth of browsers and the great influence of HTTP, many systems are implemented using Browser/Server architecture rather than Client/Server architecture. IPFS aims to create a file system that connects all computer devices. Even though IPFS proposes putting the immutable, permanent IPFS links into a blockchain transaction, we believe several elements of IPFS can be improved by combining it with blockchain. Each node in this blockchain system has a copy of the hash key for the encrypted file that is stored on the IPFS server. The goal of this system is to use the blockchain concept to promote file sharing services with greater security and lower risk.**

Keywords*—*IPFS; Blockchain; Peers; HTTP

I. INTRODUCTION

Blockchain apps interact directly with blockchains or smart contracts to reach consensus on transactions, data, or code execution. The blockchain is stored on a distributed network of heterogeneous nodes that also process transactions and, if necessary, execute smart contracts. When working with large data files, the following issue arises. Because the files aren't required for the blockchain nodes to function, the blockchain becomes bloated, resulting in data being replicated on a large number of nodes. On the one hand, the blockchain stores large files inefficiently. Files must be divided and reassembled off-chain due to block size limits. Additional data for reassembling files would also need to be saved, requiring either greater capacity or a separate system to provide the reassembly information. The data can be accessed more easily and the reassembly information can be saved if smart contracts are utilised to save file components directly. Using smart contracts to send and store large amounts of data, even in part, is expensive (for example, in terms of gas prices) and requires execution at each mining or verifying node. The cost of running the mining nodes, on the other hand, is increasing. As it moves through the network, the node must process and store additional data. As a result, mining nodes would need more bandwidth and storage space to keep the blockchain, even if just partially, resulting in higher pricing.

Blockchains have found to be ineffective for distributing and storing large files. Fortunately, file sharing services may be utilised to enable applications while keeping the blockchain small. Users can swiftly share large files while still benefiting from the blockchain. Cryptographic hashes can be given to the latter, proving that the file was available to someone at a specific time. The InterPlanetary File System (IPFS), which combines file sharing and the aforementioned hashes, is a particularly attractive file sharing platform for this purpose. IPFS employs the cryptographic hashes of the files' contents to identify, verify, and transfer them.

Similar to public blockchains, files stored on IPFS can be read and viewed by anybody who can connect to or build an IPFS node. For blockchain apps that interface with large files holding sensitive or personal information, this is a concern. As a result, this paper builds an IPFS with access control using the Solana blockchain. The modified IPFS software, termed ACL-IPFS in the following, can then connect to the smart contract and enforce the access control list's rights. ACL-IPFS allows users to register new files and grant and withdraw privileges by producing and submitting transactions to the smart contract.

HTTP gets files from a single server at a time, whereas peer-to-peer downloads data from multiple servers simultaneously. IPFS fetches data from numerous nodes at the same time, resulting in significant bandwidth savings. IPFS allows for the efficient distribution of large volumes of data without duplication, saving up to 60% for video. IPFS makes it simple to establish up robust networks for mirroring data, and files stored with IPFS are automatically versioned thanks to content addressing. Because it is one of the great equalizers in human history, the Internet has accelerated innovation, but increasing control consolidation threatens that progress. By providing technology to make the original concept of an open, flat web a reality, IPFS keeps faithful to the original vision. With or without internet backbone connectivity, IPFS enables the establishment of diversely resilient networks that provide permanent availability.

II. RELATED WORKS

Here we look /at some of the related works in the field of supply chain management using blockchain:

1. Blockchains are ineffectual for distributing and storing large data, according to the study "Blockchain-based decentralised access control for IPFS" [1]. Fortunately, file sharing services may be utilised to enable applications while keeping the blockchain small. Users can swiftly share large files while still benefiting from the blockchain. Cryptographic hashes can be given to the latter, proving that the file was available to someone at a specific time. The InterPlanetary File System (IPFS), which combines file sharing and the aforementioned hashes, is a particularly attractive file sharing platform

for this purpose. IPFS identifies, verifies, and transports files based on their cryptographic hashes. Similar to public blockchains, files stored on IPFS can be read and viewed by anybody who can connect to or build an IPFS node. For blockchain apps that interface with large files holding sensitive or personal information, this is a concern. As a result, this paper builds an IPFS with access control using the Solana blockchain. A Solana smart contract stores and dynamically modifies the access control list. The modified IPFS software, termed ACL-IPFS in the following, can then connect to the smart contract and enforce the access control list's rights. ACL- IPFS allows users to register new files and give and withdraw permissions by producing and submitting transactions to the smart contract. With each request for a file, ACL-IPFS nodes provide the public key and sign the message using a linked Solana account. This creates a relationship between the nodes and the account, allowing the nodes to use the smart contract to seek and enforce permissions.

2. "Blockchain-based secure storage and access scheme for electronic medical records in IPFS"[2]: Electronic medical records, medical photographs, diagnostic reports, infectious diseases, and other types of medical data are now more commonly employed in medical care, and a considerable volume of medical data is generated every day. Infectious diseases can not only be foreseen and planned for protection with good use of this medical data, but they can also be utilized as legal proof for doctors. As a result, it's critical to investigate how to use medical data properly. If the supplied medical data is misused wrongly, the patient's privacy will be jeopardised. As a result, keeping track of medical data access privileges is a top priority. At this time, Attribute Based encryption (ABE) is the best method for implementing access control. Medical institutions outsource encrypted medical data management to a third party (i.e., a Cloud server), which not only reduces computing expenses and saves local storage space, but also allows for speedier data retrieval.

   CP-ABE (Ciphertext-Policy-based Encryption) is a secure way to acquire quick and limited access to data. The proposed architecture, as a result of this study, not only provides for the storage of EHR data, but also for the incorporation of various associations and the security of records. When the number of qualities under policy expands in prior ABE schemes, the size of the cypher text grows quite large, and the system's overhead grows.

   Furthermore, integrating blockchain technology to modern medical settings has become a new trend, however medical data saved on the blockchain cannot be retrieved efficiently. To solve this challenge, we employ the InterPlanetary File System (IPFS) as our storage platform. IPFS is a distributed storage protocol that was created to reduce file redundancy. Each stored file is given a unique hash, allowing the user to identify the file by its hash address. Because IPFS is decentralized, there is no single point of failure.

3. "Trustworthy Electronic Voting Using Adjusted Blockchain Technology" [3]: The popular will is a well-known phenomenon for representing voter information to electoral organisations. These bodies range from college unions to parliaments. 'Vote' has evolved over time as a mechanism for conveying the will of the people when a decision must be made among multiple options. People's faith in the decisions they make by majority vote has improved thanks to the voting technology. This has undoubtedly aided in the democratisation of the voting process and the recognition of the importance of voting systems in the election of parliaments and administrations. In 2018, 167 counties out of a little over 200 have some form of democracy, whether complete, imperfect, or hybrid. People's trust in democracies is growing, therefore it's critical that they keep their faith in the vote and the voting system. The voting system arose as a platform to assist people in electing their representatives, who, in turn, create governments as a result of the growing faith in democratic institutions. The people are empowered by the power of representation because they have faith in the government to look after national security, national issues such as health and education policy, international relations, and taxation for the people's benefit.

4. “Blockchain-Based Address Alias System” [4], The goal of this article is to show how to define aliases for addresses in a human-readable manner. The key point is that solutions must be on chains rather than user-defined aliases, which are typically only saved in the user cache. As a result, new wallets, exchanges, and other applications may easily query the blockchain smart contract to acquire a list of all registered aliases. Furthermore, users will be able to utilize the same aliases throughout the ecosystem without having to do any additional work. High costs, long confirmation periods, and a lack of support for autocomplete features are some of the drawbacks of popular alias solutions. Our solution seeks to address all of the aforementioned difficulties while maintaining a decentralized and permissioned system.

III. PROPOSED SYSTEM

Our suggested solution combines Blockchain and IPFS, two fundamental decentralized networking components. We also discuss why an up-and-coming platform like Solana, rather than the more well-known Ethereum, was chosen.

A. InterPlanetary File System (IPFS)

InterPlanetary File System (IPFS) is a decentralized file sharing network that recognizes files based on their content. It employs a distributed hash table to obtain information about file locations and node connectivity (DHT). The following sample explains how IPFS works. When you submit a file to IPFS, it's split into chunks, each with up to 256 kilobytes of data and/or links to other chunks. To identify each chunk, a cryptographic hash, also known as a content identifier, is computed from its content. Content IDs are added to the previously stated links, resulting in a Merkle directed acyclic graph (Merkle DAG) that characterises the file as a whole and may be used to reconstruct any file from its pieces. A single root hash can be used to identify an entire file thanks to the Merkle DAG. The DHT is used by a node to register as a provider. In essence, the DHT is a distributed key-value store. It stores and retrieves data using node IDs and keys, both of which must be the same length, as well as a distance metric. When a node is looking for a value, it seeks for nodes that are close to the key and asks them for it. It accomplishes this by employing buckets to keep track of nodes in the network. The buckets are organized so that any network node can get precise information about its immediate surroundings. The knowledge a node has of other nodes decreases as the distance between them grows. As a result, to find the value associated with the key, a node contacts a node that is closer to the key than it is. In its answer, the latter either returns the value or points to nodes that are even closer to the key. This method is repeated until the key is found. A node locates a set of nodes that are the closest to the key and informs them of the key's value so that they can write a value. This enables nodes to swiftly write and receive data from the network. Keys, on the other hand, are only valid for a limited period of time, such as 24 hours, and must be updated frequently to stay in the DHT. It maintains track of two different types of information. A node automatically registers as a provider of the file's chunks when a file is uploaded through it. Second, the DHT comprises instructions for establishing a connection to a node specified by a certain identifier, such as an IP address. As a result, an IPFS node can utilize the DHT to connect to providers and request files. Content IDs are used by IPFS to identify and validate content. It's very useful for moving large files and bits of data. to be utilized in tandem with blockchains. The hash of Merkle DAG can be sent to the blockchain using a transaction. In IPFS, the pieces are referred to as blocks. However, it's easy to detect the difference between IPFS and blockchain blocks, the former are smaller. Chunks are referred to as blocks in this paper. As a result, the latter isn't as bloated as the former. Simultaneously, no information is available. You'll need something other than the hash to obtain the file from IPFS. This is in contrast to file sharing systems that do not use cryptographic hashes as content IDs since they require additional name resolution procedures to locate a file whose hash is in the blockchain. Furthermore, because it is difficult to create a second, distinct file with the same hash, flooding IPFS with files with the same target file identifier is impossible. To summarize, a file delivered over IPFS can easily be checked, making it impossible to deceive a user by providing several files with the same name/identifier.

B. *Solana*

Solana is a blockchain platform that is open to the public. SOL is the company's proprietary cryptocurrency. Solana uses the Proof of Stake and "Proof of History" mechanisms to establish an agreement, which the Solana White Paper argues will improve efficiency. Solana was identified as "a potential long-term challenger to Ethereum" by Bloomberg journalist Joanna Osinger in 2021, citing faster trade speeds and cost savings. The Solana blockchain went offline on September 14, 2021, because the network was split owing to a deluge of transactions, and different validators had differing perspectives on the network's condition. On September 15, 2021, the network was brought back online without warning. According to reports, Solana can process 50,000 transactions per second. It's faster than Ethereum, although there aren't nearly as many transactions at peak hours.

Solana, like many other blockchains, can run smart contracts. Melania Trump, the former First Lady of the United States, announced plans to use the platform to release non-fungible tokens on December 16, 2021. (NFTs). In a press statement, the Solana Foundation clarified that their platform choices are not formally "part of the Solanad Initiative."

IV. IMPLEMENTATION

A. *Adding a File*

IPFS creates chunks and the Merkle DAG associated with each file uploaded by a node. During this step, all content IDs associated with the file are recorded. These are then passed to the permissions package, which registers the files in the smart contract using the addFile functions. This is done by putting together the essential transactions. Transactions can also be verified using the correct content IDs. If the transactions succeed, the execution returns to the IPFS code, which saves the blocks locally and then registers as a content identifier provider. It is impossible to claim ownership of at least one of the content identifiers, and consequently the entire file, if the transaction fails. As a result, an error is returned, interrupting the execution and, as a result, the file's storage and registration as a provider in the ACL-IPFS node's storage. An adversary node that witnesses the transaction can seek to claim the same content identification, obtaining the ability to set the content identifier. A smart contract grants permission to it. Despite the fact that the hostile node's success is not guaranteed. The first transaction to be completed wins since the two transactions are competing. The addition of the second to the network makes the first obsolete. Because transactions are limited in size, it is not always possible to include all associated content IDs in a single transaction. If this is the case, multiple transactions can be used to register content identifiers. ACL-IPFS then keeps track of all of these transactions in order to report on the process' overall success.

B. *Granting Permissions*

Through the command line, ACL-IPFS allows users to simply provide access permissions to specific people. A transaction is also created and delivered to a blockchain node for this reason. Only the owner, identifiable by its Solana account, can now give or cancel a permission, because the content identity has already been registered. A condition in the smart contract ensures that this is the case, as previously stated.

C. *Retrieving a file*

A node can request and acquire the chunks corresponding to the files if the relevant permissions have been granted. It does this by locating suppliers of these chunks in the network and establishing connections with them in order to receive the chunks required to reassemble the file. The request will not be replied if the authorization has not been given or if it has already been revoked. Anyone connected to a Solana node can check a user's rights for any content identifier and make the proper judgment as a result.

D. Connect to Solana Wallet

In order to publish a block to the blockchain, a gas fee is required and for that purpose we need to establish a connection to a blockchain wallet such as Phantom (a crypto wallet). The users need to install this as an extension to their browser and create an account in it as well. When the user uploads a file to the network, a transaction is initiated and a certain amount of gas fee is deducted from the Phantom wallet automatically. Since DevNet is being used, virtual currency in any amount can be added to the wallet for smooth functioning of this project.

E. Account Creation

The number of files that can be uploaded to a single account depends on the size of memory allocated for that account by the developer. Every time that limit exceeds, an option to create a new account is provided within the application itself. Creation of a new account implies that a new keypair would be generated and assigned to the account and hence, access to the already uploaded files would be restricted on doing so. Another alternative to creating a new account is to increase the size allocated for a single account. This can be done by contacting the developer.

F. Architecture

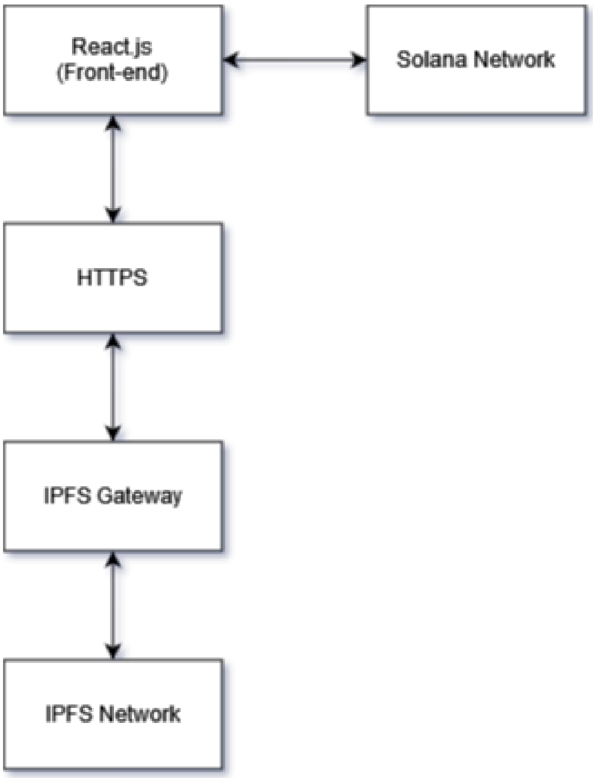

Figure 1. System Architecture

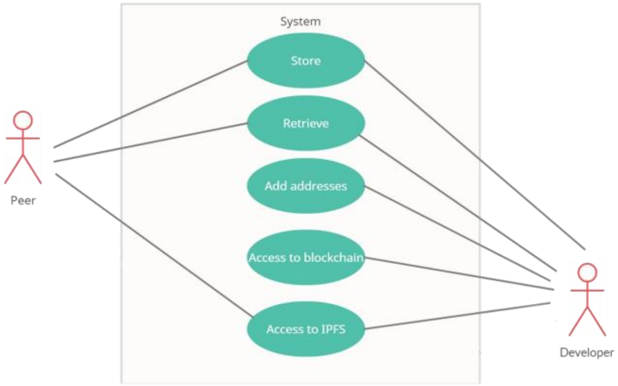

Figure 2. Use-Case Diagram

V. CONCLUSION

The purpose of this project was to solve the need for blockchain applications to exchange larger files with sensitive information within an organization. The files cannot be stored efficiently on the blockchain or uploaded using unmodified IPFS nodes, as previously mentioned. The design and development of ACL-IPFS, a blockchain-based extension to IPFS that enables access control, has been discussed for this purpose. The access control list is managed by ACL-IPFS using Solana smart contracts. Users can register files and provide or withdraw access to them using the smart contract. The IPFS software has been changed to provide access to the smart contract while also enforcing the permissions set in the access control list. The additional latency is mostly due to the interface with the blockchain, and it has been shown to be perceptible but not operationally significant.

REFERENCES

[1] M. Steichen, B. Fiz, R. Norvill, W. Shbair and R. State, "Blockchain-Based, Decentralized Access Control for IPFS," 2018 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), 2018, pp. 1499-1506, doi: 10.1109/Cybermatics_2018.2018.00253.

[2] J. Sun, X. Yao, S. Wang and Y. Wu, "Blockchain-Based Secure Storage and Access Scheme For Electronic Medical Records in IPFS," in IEEE Access, vol. 8, pp. 59389-59401, 2020, doi: 10.1109/ACCESS.2020.2982964.

[3] B. Shahzad and J. Crowcroft, "Trustworthy Electronic Voting Using Adjusted Blockchain Technology," in IEEE Access, vol. 7, pp. 24477-24488, 2019, doi: 10.1109/ACCESS.2019.2895670.

[4] Bodziony, Norbert, Paweł Jemioło, Krzysztof Kluza, and Marek R. Ogiela. 2021. "Blockchain-Based Address Alias System" in Journal of Theoretical and Applied Electronic Commerce Research 16, no. 5: 1280-1296. https://doi.org/10.3390/jtaer16050072