Integrity Management and Access Control of External Storages using Blockchain Technology

Hasan Mohammad Shahriar
Research and Development
Kona Software Lab
Dhaka, Bangladesh
h.m.shahriar@konasl.com

Muhammad Nur Yanhaona
Research and Development
Kona Software Lab
Dhaka, Bangladesh
nur.yanhaona@konasl.com

***Abstract*—**

***Index Terms*—peer-to-peer computing, distributed information systems, document handling**

I. Introduction

Since its inception in 2008 [11], the blockchain technology has gained widespread attention as a transformative technology that can revolutionize many industries [1]. Blockchain based digital currencies such as Bitcoin [11], Ether [17], and Ripple XRP [6] are already being considered viable alternatives to existing currencies in trade and commerce for their security and ease of transfer. Blockchain smart contracts [14] [17], on the other hand, have spawned innovative applications in many business and financial sectors due to their capacity of encoding the rules of interaction and ensuring their enforcement.

Central to the appeal of blockchain technology is its maintenance of a distributed ledger of transactions – called the blockchain – in a peer-to-peer network of autonomous and anonymous entities. In a blockchain network, all entities are even and none of them is trusted; still, the security and integrity of the transaction ledger can be guaranteed. This feat is achieved by a complete replication of information in all network participants where each participant validates and executes all transactions. As long as the majority of the network participants are honest, the outcome of the transactions, i.e., the state of the blockchain ledger can be trusted [12].

The blockchain technology's decentralization of trust through information and processing replication in a theoretically infinitely scalable peer-to-peer network is leading innovations and renovations in many application domains where trust and information security are key concerns. However, problem in one area in particular appears to be a major obstacle for blockchain based application adoption. This is the problem of document storage.

The blockchain technology is inherently unsuitable for storing bulky information such as files and media contents due to the networking and storage cost associated with their management. Peculiarities of blockchain ledger maintenance such as *blockchain reorg* [2] further complicates the situation by making direct integration of existing trusted storage solutions with a blockchain network difficult. Finally, public blockchain technologies find the continual preservation and integrity insurance requirement for trustworthy document storage in conflict with their blockchain transaction ledger maintenance incentive where participants are only being paid for extending the ledger of transactions[1] and they can join or leave the network at any time.

Nevertheless there are some blockchain based or blockchain inspired storage technologies such as Ethereum Swarm [15], Filecoin [8], Storj [16], and IPFS [5] already available. These solutions break down a user's file into a series or hierarchy of data chunks then distribute the chunks to the peer-to-peer network. On a broad level, some of these storage solutions are like traditional distributed hash tables [9] [7]. Some others are like peer-to-peer file sharing services such as the popular Bittorrent [13]. These solutions apply some bitcoin-like incentive mechanisms on top of these base technologies to motivate the network participants to retain and serve data chunks upon users' request.

The motivation for these solutions is that they protect the users from vendor locked-in and they offer an overall larger storage capacity compared to existing storage alternatives. However, blockchain based solutions have the common problem that the owner (or user) has to take the responsibility of ensuring persistence and integrity of his/her data in the blockchain by retaining document metadata and issuing periodic audits. Furthermore, despite the combined storage capacity being huge, the download bandwidth can be significantly low as the network peers may be running simple commodity hardware behind low-speed network connections. In addition, designing incentive mechanisms for long-term persistent of documents in a mining based blockchain network is difficult

Legacy databases of existing applications are also an obstacle for the applications' migration to the blockchain domain. Data stored in proprietary data centers are often confidential that a typical administrator may not be comfortable to put in the hands of anonymous blockchain participants. Further, when existing cloud storage providers [10] [3] have already solved the storage capacity, scalability, and cost-effectiveness

This research is funded by KONA Software Lab, Limited.

[1] by mining transactions into new blocks

problems for the clients; there is little motivation for moving data into a blockchain storage.

We believe, to steer blockchain application innovations, blockchain technology should be supportive of existing storage solutions instead of being their competitors. In other words, the goal should be integrating existing storage technologies with blockchain – not toppling them. A collaboration of technologies can bring the best of the both worlds. The blockchain technology can ensure integrity of external documents and control access to them according to the transparent governance of blockchain smart contracts and leave the actual storage, delivery and capacity scaling to a matured storage technology. Here, the blockchain technology is ideally suited for its part because information corruption in a blockchain network is very difficult and access rules written in the smart contracts are self-enforcing if integrated properly.

In our scheme, location, signature, access control configuration, upload/download fees and so on metadata information about the document is stored in blockchain smart contracts. A user gets access to the externally stored documents by interacting with a *Storage Integration Blockchain Gateway* from a blockchain client application. Any conversation with the gateway involves a series of transactions in the blockchain network and happens following the instructions of some secure interaction protocol. Finally, if the gateway approves the access request then it generates an access token to the external storage that the user uses to upload/download a document with the external storage directly. In case of a download, in particular, the client application verifies document authenticity by locally computing the document signature and matching it against the signature stored in the blockchain before delivering the document to the user.

Figure I depicts a high-level description of the system architecture of our solution. Observant readers will notice that

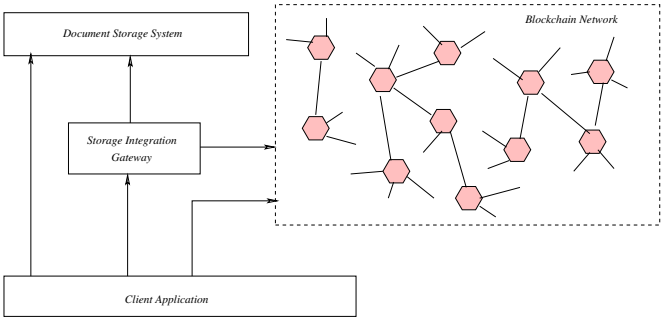

Figure 1. External Storage System Integration Model

the *Blockchain Network* and the back-end *Document Storage* of Figure I can scale up to meet users' high-availability and other quality of service needs. However, the same cannot be said about the *Gateway*. If proper care is not taken, its failure can make documents unavailable for client access. We avoided this grave potential problem by ensuring that the gateway database only contains information derived from the blockchain network. Hence, the gateway can be limitlessly replicated and the same back-end document storage may be interfaced by many gateways at the same time. Furthermore, we support different gateways to be configured differently to charge users for upload and/or download differently based on their quality of service (QoS) and application requirements.

The underlying core innovations that make our solution work are as follows:

1) Efficient, secure, and accountable document upload-download protocols that support configurable blockchain based payments.
2) A generic document access control configuration paradigm using blockchain smart contracts.
3) Enforcement of access control rules in the storage integration gateways.
4) Fault-tolerant design of the storage integration gateway against blockchain transaction reversal and back-end document storage failures.

This paper describes these core innovations and discusses some associated concerns. The rest of the paper is organized as follows:

A. Paper Organization

Section II elaborates on the scope of the external storage integration problem in our modeling, Section III describes document upload/download protocols and analyzes their characteristics, Section IV presents our innovation on blockchain smart contract based document access control policy configuration and its enforcement, Section V examines some gateway design concerns, Section VI discusses some related work on blockchain based/inspired document storage, Finally, Section VII concludes the paper.

II. The Scope of the Storage Integration Problem

To discuss the scope of the external storage integration problem, readers first need to understand our vision of responsibility breakdown for blockchain powered applications requiring document storage. For this latter discussion we refer to the model of Figure I.

We envision that the application logic of a blockchain powered application (subsequently referred as a *Blockchain Application*) will be stored in the blockchain network in the form of blockchain smart contracts. Information update, payment processing, and any auditing initiated by users' access to the blockchain application will be governed by those smart contracts and reflected as transactions in the blockchain ledger. If we categorize aspects of users' interaction with a blockchain application into *4–As*: *Authentication* of user, *Authorization* of request, *Access* to information, and *Audit* of change; all *4-As* are handled by the blockchain network – except for the case of access/updating documents such as files and images.

Such documents will reside on one or more external document storage systems. At present, there is no convincing solution for reliable interaction of a blockchain network with an external information system in the literature. Hence *4-As* related to document access/update cannot be handled by the blockchain network as being done for other user interactions.

A feasible alternative is that a user's blockchain identity and information in the blockchain ledger set the rules for the user's access to the external storage systems. Then a gateway service interacting with both the blockchain network and the storage systems enforces the *4-As* on behalf of the blockchain network.

To elaborate, a user will identify him/herself with the gateway with his/her blockchain identity and request upload-/download of a document in an external storage as additional information associated with some blockchain smart contract. The gateway will check if the blockchain ledger contains permission information that support authorization of the user's request. Then the user and the gateway undergo an access authorization protocol that results in several blockchain transactions for auditing and payment processing. If authorization is successful, the gateway creates a restricted session with the external storage that the user uses to upload/download documents to/from the external storage.

The aforementioned breakdown of responsibilities for external storage access limits the scope of the storage integration problem to two primary activities:

1) development of a storage access permission control mechanism that exclusively uses blockchain information, and
2) designing secure, reliable, and accountable protocols for paid or free document upload/download with external storages.

Note that although we accept the blockchain network as the veritable source of information for both activities, the storage integration gateway needs to consider that any transaction in the blockchain can be reversed due to blockchain ledger reorganization [2]. Consequently, all access control decisions must be made based on the latest state of the blockchain ledger and the upload/download protocols should support rollback and resume. Doing this elegantly is a major design concern for the gateway.

Involvement of the blockchain network in document upload and download with external storages provides a natural mechanism for document integrity checking. During a document upload, a short and unique document signature can be generated from the document content[2] and stored in the associated blockchain smart contract. During a download, the client application can recompute the signature from the downloaded content and match that with the signature found in the blockchain smart contract. If the two signatures do not match then the document has been modified or corrupted outside the guidance of the blockchain network and the client rejects the document. This simple scheme of using blockchain ledger's immutability to ensure document integrity has been used by others also [4].

Subsequent sections chronologically describes the upload and download protocols, the permission control mechanism, and the gateway design.

[2]for example, a hash of the document byte stream can be the document signature

References

[1] Break through with blockchain. https://www2.deloitte.com/us/en/pages/financial-services/articles/blockchain-series-deloitte-center-for-financial-services.html. Accessed: 2019-05-06.

[2] Chain reorganization. https://en.bitcoin.it/wiki/Chain_Reorganization. Accessed: 2019-02-12.

[3] Google cloud storage: Features and benefits. https://cloud.google.com/storage/features/. Accessed: 2019-05-15.

[4] Seamless blockchain certification now even easier. https://stamp.io/. Accessed: 2019-05-24.

[5] Juan Benet. Ipfs - content addressed, versioned, p2p file system, 07 2014.

[6] D. Stalin David. The ripple protocol consensus algorithm. 2014.

[7] Michael J. Freedman and David Maziéres. Sloppy hashing and self-organizing clusters. In M. Frans Kaashoek and Ion Stoica, editors, *Peer-to-Peer Systems II*, pages 45–55, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.

[8] Protocol Labs. Filecoin: A decentralized storage network. https://filecoin.io/filecoin.pdf, 2017.

[9] Petar Maymounkov and David Mazières. Kademlia: A peer-to-peer information system based on the xor metric. In *Revised Papers from the First International Workshop on Peer-to-Peer Systems*, IPTPS '01, pages 53–65, London, UK, UK, 2002. Springer-Verlag.

[10] James Murty. *Programming Amazon Web Services*. O'Reilly, first edition, 2008.

[11] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. *Cryptography Mailing list at https://metzdowd.com*, 03 2009.

[12] Rafael Pass, Lior Seeman, and Abhi Shelat. Analysis of the blockchain protocol in asynchronous networks. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *Advances in Cryptology – EUROCRYPT 2017*, pages 643–673, Cham, 2017. Springer International Publishing.

[13] Johan Pouwelse, PawełGarbacki, Dick Epema, and Henk Sips. The bittorrent p2p file-sharing system: Measurements and analysis. In *Proceedings of the 4th International Conference on Peer-to-Peer Systems*, IPTPS'05, pages 205–216, Berlin, Heidelberg, 2005. Springer-Verlag.

[14] Nick Szabo. Formalizing and securing relationships on public networks. *First Monday*, 2(9), 1997.

[15] viktor trón, aron fischer, dániel a. nagy, zsolt felföldi, and nick johnson. swap, swear and swindle incentive system for swarm. https://swarm-gateways.net/bzz:/theswarm.eth/ethersphere/orange-papers/1, May 2016.

[16] Shawn Wilkinson, Tome Boshevski, Josh Brandoff, and Vitalik Buterin. Storj a peer-to-peer cloud storage network. https://storj.io/storj2014.pdf, 2014.

[17] D. Wood. Ethereum: a secure decentralised generalised transaction ledger. 2014.