NFB: A Protocol for Notarizing Files over the Blockchain

Haikel Magrahi
Paris VIII University
IRT SystemX, Paris-Saclay, France
Student and Data Engineer
Email: megrhi.haikal@gmail.com

Nouha Omrane
Docapost DPS, France
IRT SystemX, Paris-Saclay, France
R&D Expert
Email: nouha.omrane@docapost.fr

Olivier Senot
Docapost DPS, France
Director of new services
Email: olivier.senot@docapost.fr

Rakia Jaziri
LIASD, Paris VIII University
Associate professor
Email: rjaziri@ai.univ-paris8.fr

***Abstract*—Blockchains, such as Bitcoin and Ethereum and their respective P2P networks have seen significant adoption in many sectors in the past few years. All these technologies that use the Blockchain pattern show that it is possible to rebuild any transactional system with better performance without relying on any trusted parties to manage transactions between peers. This insight has lead many companies to invest millions to understand the technology and to find a way to migrate from centralized to decentralized solutions.**

These solutions need to store large amounts of data in a secure and confidential way. Many distributed storage systems now exist using Blockchain technology. But none of the tools proposed are able to manage the document life cycle nor archive documents based on regulatory compliance.

In this paper, we describe our protocol named NFB (Notarizing Files over the Blockchain). This protocol ensures the communication between two systems: a permissive Blockchain and a secured centralized archiving document management system. The method described is used to allow users to archive, control, analyze and validate their transactions in a system that offers confidentiality, security and distribution features.

Keywords—Blockchain, Archiving documents, Regulatory compliance, Protocol.

I. Introduction

Blockchain, initially, gained traction in the financial sectors slowly, by offering trading, exchanging, supply-chain services securely and efficiently without needing any central point of control. Then, cryptocurrency blockchains and their respective P2P networks started to be used beyond exchanging money which leaded various industries to consider exchanging objects, not just cryptocurrencies, and building decentralized application introducing the Blockchain 3.0.

Today, Blockchain technology is used in many sectors such as media, healthcare, education, etc. More precisely, this technology is used along with other applications such as File Storage or Document Time-stamping.

In this paper, we are interested in the data archiving field. Recently, many solutions have been released such as Storj, IPFS [1], Blockstack [2], etc. Despite their decentralized approach, they still require a set of rules to ensure the long term storage of documents and their integrity. Moreover, the decentralized model poses some challenges when there is a need to verify user identity or to ensure document access confidentiality.

A document archiving management tool is therefore important for the implementation of sustainable operations related to the management of document life cycle in order to archive documents in compliance with legal requirements. In this paper, we present our protocol NFB which relies on a centralized archiving document management solution called OKORO which is certified NF 461 and ISO 14641-1.

II. Related work

Since the creation of the Bitcoin Blockchain [3] in 2009, many types of Blockchain has been developed. Some Blockchains are designed to be completely public and they expose the ledger to all participants, such Bitcoin and Ethereum. Other Blockchains are permissioned and all information is encrypted in a distributed ledger, such as Quorum and Hyperledger [4]. Blockchains 2.0 expose a collection of business rules called smart contracts which are deployed in the network, distributed and validated by all the peers in the network. A smart contract can be highly serviceable in automating business processes in a trusted way by allowing all stakeholders to process and validate contractual rules as a group which increases confidence and security. In this section, we compare the best-known Blockchain technologies and detail the best-known storage tools in P2P network Blockchains.

A. Ethereum

Ethereum is the second most popular public Blockchain for exchanging assets and payment within the Blockchain 2.0. Ethereum supports smart contracts and offers decentralized applications. Each transaction is paid with cryptocurrency called ETH, and has a Gas. To validate the transactions, Ethereum supports Proof of Work: a consensus-based on solution using Miners.

978-1-5386-3662-6/18/$31.00 ©2018 IEEE

B. Quorum

Quorum [5] is an Ethereum fork and a private Blockchain. The founders modified Ethereum by adding permissions, private transactions and they changed the consensus from the POW to a voting system, which massively boosted the transactions per second rate.

C. Hyperledger

Hyperledger is a permissioned shared ledger that offers high degrees of confidentiality, resiliency, flexibility and scalability hosted by Linux Foundation. Hyperledger supports smart contracts named chaincodes and channels which are sub-blockchains. It doesn't support cryptocurrencies nor gas mechanisms. Hyperledger uses PBFT consensus by separating the validation into different roles (e.g. Commiter, Endorser, etc.).

The distributed ledger contains all validated transactions made between peers in the P2P network. But the data exchanged between users (e.g. files) are not stored in the ledger. For that reason, numerous storage solutions and protocols have been created during the evolution of this pattern offering more secure and decentralized storage solutions without relying on any central control point. In the following paragraphs, we illustrate the best-known storage tools in P2P network Blockchains.

D. Storj

Storj [6] is a Blockchain-based cloud storage solution. It offers client-side encryption which increases security. Also, it offers the possibility of transferring and sharing data without depending on a central point of control. More precisely, Storj secures documents by encrypting and splitting them into shards. This mechanism is similar to BitTorrent.

E. IPFS

IPFS [1] is a Blockchain based File-system that is distributed across all peers in the network. IPFS combines Git, Bit-Torrent and the Self-Certified File-systems. IPFS attributes a cryptographic hash to each file. Then, it removes duplicated files and gives them a version by using Git mechanism. Next, each network node stores a copy and some indexing information for search optimization.

F. Blockstack

Blockstack [2] is a new decentralized internet where users keep their data encrypted before backing it up in the cloud. This eliminates the need for blind trust in third parties and makes it easier to keep your data safe. Applications run locally. In fact, they are loaded via a secured domain name system and exist on your device.

As we discussed so far, all these solutions involve storing documents, by replicating and distributing them across the network. Applying this technique increases availability and performance. But in the other hand, it renders documents less secure and more exposed in different nodes in the network. Furthermore, archiving involves more than storing or saving data. Documents must be processed and validated in order to define their life cycle and the data must be stored in places known by clients. Moreover, the most of the time, archiving tools should be legally compliant which is not the case for these blockchain storage solutions.

In the next section, we describe our protocol NFB that is built on top of the Quorum Blockchain [5] and uses OKORO as a centralized DMS.

III. NFB Protocol

Notarize File over the Blockchain Protocol (NFB) guarantees communication between two different ecosystems: the Blockchain and a centralized archiving document solution (DMS). It offers major services such as document archiving, document retrieval and document proof of existence thanks to the use of a DMS for archiving documents and the Blockchain to trace transactions related to the archived documents.

A. Archiving documents

This service accepts encoded documents associated to a collection of metadata that represent a set of information (e.g. title, date, etc.). Documents and their metadata are stored and indexed in the DMS. The process controls and validates document format and its metadata in the DMS side. Then, NFB traces all transactions related to these actions in the Blockchain. Several items of information are traced in the distributed ledger. Some information is related to the transfer process (e.g. public address of the sender, transaction timestamp, etc.). Other information corresponds to the data archiving system such as document ID, document hash, file name, etc.

B. Retrieving documents

Besides archiving documents, NFB offers the possibility of retrieving documents in order to enable searching and downloading documents physically. Only users that have the right to search and access a set of documents can operate these actions. The retrieving process accepts one or several keywords that correspond to user queries. Keywords are related to either document metadata such as date, document hash, information that are traced in the Blockchain such as document ID or the transaction label.

The search result corresponds to one or several documents. Users can download the selected documents, only if they have the right to operate this kind of action. At the same time, NFB traces these actions in the Blockchain. Transactions, that are traced in the distributed ledger, contain several items of information such as document ID, timestamp of the retrieval action, etc.

C. Document proof of existence

The protocol NFB enables auditing documents publically in order to prove their existence through time. If someone wants to verify a document integrity and existence, the protocol offers the possibility to compute the document hash and verify the document existence thanks to the correspondence between items of information that are permanently traced in the Blockchain (e.g. document hash) and those related to a document archived in OKORO.

More precisely, NFB offers a real-time streaming process for transactions and blocks that are validated in the Blockchain for those who want to monitor transactions in the Blockchain. Users can easily search for transactions, related to a document's proof of existence, by quering the Blockchain using keywords like document hash, timestamps or by transaction hash.

Users can select a transaction from the resulting output list. Then, they can check items of the document information traced in the Blockchain (e.g. document hash) and other items indexed in OKORO (e.g. document title). This proves that a document was well archived. Otherwise, NFB matches zero transactions with the user query.

In the next section, we go deeper into our NFB protocol by exploring its architecture and the modules that make it easy to maintain and to extend.

IV. NFB ARCHITECTURE

The implementation of our protocol NFB relies on the micro services architecture standard. Figure 1 describes the major modules of our prototype. The first module defines all Blockchain dependencies such as the SDK, smart contracts' source code, etc. Also, it implements all the functionalities that enable the writing and reading in/from the Blockchain such as deploying smart contracts, sending transactions between peers and even retrieving blocks.

The second module, named ”OKORO layer”, includes all DMS dependencies. It defines a set of web services to manage documents through a centralized document archiving solution (named OKORO). For example, such functionnalities are opening a stream, injecting documents and retrieving documents.

The third module plays the role of a gateway. It synchronizes and orchestrates the tasks between the Blockchain layer and the DMS layer. Moreover, it manages authentication and authorization, returns web UI for users to test the protocol and defines session management.

Using this modular application makes our protocol easy to reuse and facilitates source code updates by offering a set of interfaces that you need to re-implement it in order to change the Blockchain technology or the used DMS.

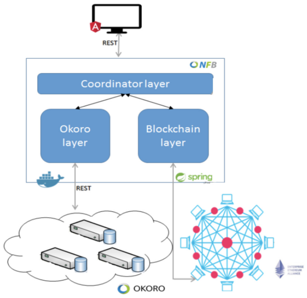

Fig. 1. NFB architecture.

A. Proof of Archivability

As we explained in the NFB Protocol section, this service archives documents in a centralized trusted archiving solution and traces these action requests and responses in the Blockchain (figure 2).
(1) Coordinator layer decodes documents and sends them with their metadata to the DMS layer
(2) The DMS layer establishes a connection to the centralized solution and starts the archiving process.
(3) The coordinator layer establishes a connection to the blockchain based on the user credentials.
(4) The Blockchain layer uses a specific SDK or a set of custom web services to trace the action and other information in the distributed ledger.
(5) The DMS layer forwards the response to the coordinator layer and reprocesses the same step (4) to trace the response action into the Blockchain.

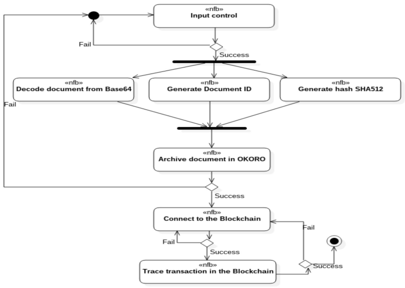

Fig. 2. Archivability service activity diagram.

B. Proof of Retrievability

As we explained in the NFB Protocol section, this service offers the possibility of searching and downloading documents from a centralized trusted archiving solution and also tracing request and response actions in the Blockchain (figure 3).
(1) The coordinator layer loads the keyword introduced by a user and sends a search request that contains a set of keywords to the DMS layer.
(2) The coordinator layer establishes a connection to the centralized archiving solution (OKORO) and starts retrieving the related documents by processing a set of web services. At the same time, the coordinator layer (3) establishes a connection based on the user address by sending a request to the Blockchain layer.
The Blockchain layer (4) uses a specific SDK to connect to the Blockchain or a set of custom web services to trace that action with some other information in the Blockchain. When the downloading process terminates (7), The DMS layer will format and forward the response to the coordinator layer. If a user gets the response, it will reprocess the same steps (3,4,5,6) to trace the response action into the Blockchain.

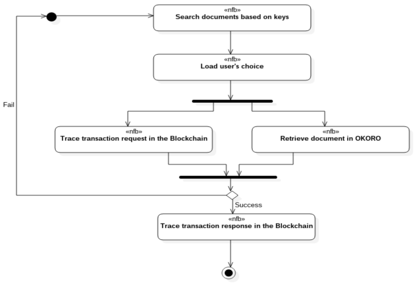
«nfb»
Search documents based on keys
«nfb»
Load user's choice
Fail
«nfb»
Trace transaction request in the Blockchain
«nfb»
Retrieve document in OKORO
Success
«nfb»
Trace transaction response in the Blockchain

Fig. 3. Retrieving service activity diagram.

C. Proof of Existence

As we explained in the NFB Protocol section, this is the most important service in our protocol. It proves a document's existence and integrity by validating the existence of a document transaction in the distributed ledger. It means that this service provides a transaction related to a document that already exists in OKORO. In other words, POE selects all metadata related to a document within the transaction and searchs for it in OKORO. When the user wants to prove the existence of a document, NFB traces that action in the Blockchain too. This service includes many search possibilities such as user address, document hash, document title, action label or even by loading a document (to calculate its hash).

Figure 4 shows the activity diagram related to the document proof of existence service.

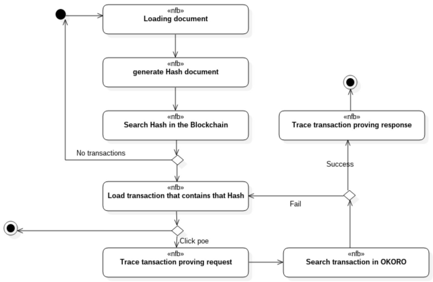
«nfb»
Loading document
«nfb»
generate Hash document
«nfb»
Search Hash in the Blockchain
«nfb»
Trace transaction proving response
No transactions
Success
«nfb»
Load transaction that contains that Hash
Fail
Click poe
«nfb»
Trace tansaction proving request
«nfb»
Search transaction in OKORO

Fig. 4. Document proof of existence activity diagram.

In the next section, we detail the technologies used to implement the protocol NFB.

V. Experiment

In this section, we describe technologies that were chosen in the implementation of NFB protocol.

A. Used technology

OKORO is a document archiving solution (DMS) introduced by Docapost DPS as a tool for archiving documents with regulatory compliance. This solution offers many services such as archiving documents, document life cycle management and instant document search. All the proposed web services are secured (full traceability, optimized backup of sensitive documents, etc.).

On Blockchain technology,our first choice was to use the Hyperledger Blockchain as the perfect choice in our PoC by offering a real database as a ledger which ameliorates storage capacity and quering. But, the underdevelopment of hyperledger and the lack of a stable version was a real problem. That's why we opted to use the Quorum Blockchain as a P2P network to trace transactions related to documents archived in OKORO.

Technologies used in our PoC were chosen by considering the richness of the community, the maturity of the technology and its performance especially Java 8, Java Spring, JavaScript and Docker.

B. Results

We deployed and tested NFB Protocol in a VM Openstack using Ubuntu server with 4vCPU, 8GB RAM, and 30GB storage. Our solution offers both web-UI and web services endpoints to be integrated as services with other tools.

We tested our protocol as a module integrated in a use case related to a solution proposed by PSA group in the automobile sector. In this use case, users can archive their documents using NFB and verify a document's existence publically for audit purpose.

This demonstrates an easy integration of our PoC with existing solution and good results by guaranteeing archiving and retrieving documents and tracing the corresponding transactions in the Blockchain. NFB protocol traces all the archiving/retrieving actions in the Blockchain and archives the physical documents in OKORO.

Conclusion

NFB is a protocol that guarantees communication between a centralized document archiving solution and the Blockchain in order to notarize files over the Blockchain. NFB offers three major services: archiving documents, retrieving documents and document proof of existence. Using NFB allows users to prove publically that a document exists (or not) by returning its metadata indexed in the secured archiving solution and based on information traced in the Blockchain, during a period.

NFB was integrated in the PSA Peugeot [7] use case to archive invoices and to trace users' actions in the Blockchain. NFB is a protocol that can be integrated easily with any given centralized DMS solution (e.g. Alfresco, SAP DMS, etc.) and with any given Blockchain-based solution (e.g. Hyperledger, Ethereum, etc.).

References

[1] J. Benet, "Ipfs - content addressed, versioned, p2p file system," *CoRR*, vol. abs/1407.3561, 2014.

[2] R. S. Muneeb Ali, "Blockstack: A new decentralized internet," *http://blockstack.org*, 2017.

[3] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system,? http://bitcoin.org/bitcoin.pdf."

[4] L. Foundation, *Hyperledger: Whitepaper*. http://www.the-blockchain.com/docs/Hyperledger-Whitepaper.pdf.

[5] J. Morgan, *Quorum: Whitepaper*. https://github.com/jpmorganchase/quorum-docs/blob/master/Quorum-Whitepaper-20v0.1.pdf.

[6] S. Wilkinson, T. Boshevski, J. Brandoff, and V. Buterin, "Storj a peer-to-peer cloud storage network," 2014.

[7] K.-L. Brousmiche, T. Heno, C. Poulain, A. Dalmieres, and E. Ben-Hamida, "Digitizing, Securing and Sharing Vehicles Life-cycle Over a Consortium Blockchain: Lessons Learned," in *IFIP NTMS Blockchain and Smart Contracts Workshop (BSC18)*, 2018.

ACKNOWLEDGMENT

This research work has been carried out under the leadership of the Institute for Technological Research SystemX, and therefore granted with public funds within the scope of the French Program Investissements d'avenir. This paper has been accepted as a short paper.