Notara: Efficient Blockchain Asset Notarization Service

Guillermo Toyos-Marfurt, Rouwaida Abdallah, Álvaro García-Pérez
Université Paris-Saclay, CEA, LIST
{firstname.lastname}@cea.fr

***Abstract*—Traditional asset notarization methods are inefficient, expensive, vulnerable to fraud, and rely on the trust of a third party, such as a notary. In contrast, blockchain-based smart contracts enable the creation of decentralized notarization services, ensuring transparency and security without intermediaries. However, existing blockchain-based solutions often lack the flexibility for specific deployment for individual entities our groups. This work presents a demonstration for a unique customer-deployable notarization service, based on blockchain smart contracts. Our solution is designed to be a cost-efficient notarization service that can be quickly integrated into decentralized applications requiring notarization functionalities.**

***Index Terms*—Blockchain, Smart Contract, Notarization, Decentralized Application**

I. Introduction

Through its decentralized and immutable properties, blockchain has become a promising force in the development of novel applications and technologies across various sectors. Traditional methods of asset notarization are often cumbersome, time-consuming, expensive, vulnerable to fraud, and they rely on centralized authority. Blockchain technology offers a solution to these problems by providing a tamper-proof decentralized ledger for verifying and recording data in a secure and transparent manner [4], [13].

Blockchain notarization involves writing a signature of the digital file, associated to the document to be notarized, in the distributed ledger. This ensures the authenticity of the document and creates a timestamp for it, making an unambiguous record of the existence of a document at a specific moment in time. This provides valid proof in case of disputes without revealing the content of the document [6]. The immutability of the blockchain can reduce fraud more efficiently than traditional notarization technology [5].

Moreover, the transparency and accessibility of blockchain networks ensure that notarizations are available to all relevant parties at any time. It also eliminates the need for intermediaries, and enables direct peer-to-peer verification. However, existing blockchain-based notarization solutions often lack the flexibility to be deployed for specific customer needs or requirements, as they are typically designed as standardized platforms that limit customization and control. These solutions require users to depend on the service providers' infrastructure, which can reduce the adaptability and increase the costs. In response to these limitations, we propose a decentralized notarization service that enables customers to independently deploy and customize their own notarization systems.

We present a demonstration of a notarization system, deployable as a decentralized application (DApp) using smart contracts. Smart contracts are programs deployed and executed on a blockchain network, which allow the creation of applications that inherit the properties of blockchain.

This work is organized as follows: First in Section II we overview the related work. Then in Section III we present a selection of real usecases for blockchain notarization. Next, in Section IV we describe our solution and we analyse its main characteristics. Finally, in Section V we give a brief conclusion of the proposed system.

II. Related Work

Several blockchain notarization systems have emerged on public blockchains like Bitcoin and Ethereum.

The OpenTimestamps project provides a blockchain notarization service using the Bitcoin network [8]. A file can be uploaded to the OpenTimestamps' web application, which returns a proof of notarization containing, among other data, the hash of the file. Internally, this hash is sent to an OpenTimestamps server, which periodically commits transactions to the Bitcoin network by placing all pending timestamps in a Merkle tree.

Similarly, the Proof of Existence project [11] also leverages the Bitcoin network for providing notarization features. It provides a deployable web application that allows users to create special transactions on the Bitcoin network, with some arbitrary data appended to it, while burning a small amount of bitcoin in the process[1]. However, the suitability of using Bitcoin for storing arbitrary data is debated. Notarizing data on the Bitcoin blockchain requires burning bitcoin, which is a finite resource, while the demand for data storage continues to grow indefinitely [10]. In contrast, smart contract-capable blockchains like Ethereum are considered better suited for data notarization, as they are able to store arbitrary data by design.[2]

Palmisano et al. conducted a comparative study of existing blockchain notarization solutions [9]. Multiple variations exist,

[1] As of September 2024, the minimum bitcoin input required for such a transaction is 0.00025 BTC

[2] In practice, the amount of data that can be stored is limited by the transactions fees. For instance, writing a byte of data in a smart contract costs 625 gas [12], equating to approximately 0.01 USD at current Ether prices on the Ethereum Mainnet.

each with its own characteristics, file management methods, and specific blockchain networks. However, many of these solutions are not straightforward to integrate in a system or rely on third parties.

Merkle trees are a well-established cryptographic technique for generating a single hash representing a set of files. Each file is hashed individually, and these hashes are recursively combined in pairs until a final root hash, known as the Merkle root, is obtained. This root hash succinctly verifies the integrity of the entire set of files, enabling efficient detection of any changes within the set.

Unlike existing blockchain-based notarization solutions, which are often rigid and dependent on external service providers, there is a need for more flexible, customer-deployable systems that allow customers to maintain full control over their notarization process.

III. Use Cases

The starting point of our notarization system is the Blockchain Verte project [7], which aims to provide low-energy blockchain solutions for data notarization and auditability, specifically for climate-related data. Here, a network of CO2 sensors is deployed in the urban area of Paris to collect environmental data, which is then processed by the climate laboratory that deployed the sensors. This data collection and processing pipeline is securely notarized on a blockchain to ensure that the data remains trustworthy and verifiable by the public, which ranges from researchers to ordinary citizens. The notarization is accomplished by producing a Merkle tree, where each file needed to process the data is hashed. Hashes are then combined to obtain the Merkle root. The Merkle root is then stored on the blockchain. This approach minimizes the number of blockchain transactions, reducing energy consumption, which aligns with the Blockchain Verte's emphasis on environmental sustainability. An user can later audit and verify the data by downloading all the raw files used for processing the data from the concerned laboratory via a dashboard available within the system, and independently re-executing the data processing. The blockchain ensures the integrity and authenticity of the data, holding the laboratory accountable of any inconsistencies or tampering.

In our approach, called Notara, this notarization scheme is generalized beyond the specific use case of the Blockchain Verte project. Unlike existing solutions that often depend on a centralized or shared infrastructure, Notara allows each project, organization, or use case to deploy its own independent notarization system. This flexibility enables diverse applications, such as notarizing documents and certificates in supply chains (as in the Digital Product Passports (DPPs) [2]), securing intellectual property through timestamped proofs of ownership, ensuring the integrity of medical and research data for regulatory compliance and reproducibility, verifying digital identities and credentials, recording energy consumption and carbon emissions data for environmental reporting, and notarizing digital contracts and agreements.

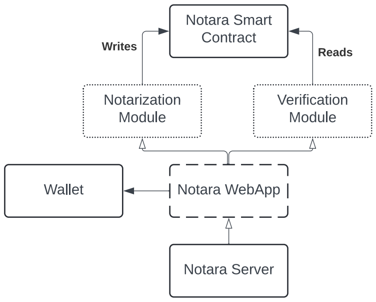


Figure 1: Diagram of the Notara system architecture, showing components, communication pathways (dark-tipped arrows), and component compositions (white-tipped arrows). Dashed boxes indicate virtual components served by the Notara server.

IV. Design and Implementation

Notara follows a conventional DApp architecture [3] composed of a server hosting a web application that provides an interface for notarization and verification, a smart contract deployed on a blockchain network, and the user's wallet. Figure 1 illustrates a diagram of this architecture.

The Notara server is only responsible for serving the files comprising the Notara Web Application. The web application connects to the Notara smart contract using the user's wallet, which holds the user's credentials on the blockchain network.

The system is customer-deployable. Any user with a compatible wallet can deploy the Notara smart contract and interact with it using the Web App. The Notara smart contract address is a parameter in the web application. Users do not need to trust the server for document verification, as there is no communication between the server and client apart from serving the DApp files. For added security, clients can host their own Notara server. The system relies on the integrity of the blockchain network rather than trusting a third party.

The demo implementation includes a smart contract written in Solidity, hosted on the Ethereum blockchain network, and a web application connecting to a MetaMask wallet using the Ethers.js library.

The web application has two core functionalities: notarizing a set of files by writing into the smart contract, and verifying if a file has been notarized. For notarization, the user uploads a set of files and optionally provides a description. The web application then builds a Merkle tree of the files and writes the tree's root hash into the smart contract. This hash is written in an indexed array, the index is returned to the user, allowing they to reference a notarization event and its description.

Since each hash is generated from the file's content, it remains independent of metadata, such as the filename. Moreover, as all files are notarized in a single transaction, their upload order to the web client does not affect the Merkle tree's structure, since the leaf order can be determined by an arbitrary rule.

To verify if a file has been notarized, Notara implements two methods:

1) The user uploads the set of notarized files and the index. The client builds the Merkle tree and compares its root against the hash stored in the smart contract using the index. If the notarized hash matches, the documents are valid.
2) The user provides the file's hash (a leaf in the Merkle tree), the root hash, and a proof composed of the intermediate hashes used to get the Merkle root starting from the file's hash. The web client verifies the proof's integrity and the root hash's notarization. If the proof is correct and the root hash matches, the document is valid.

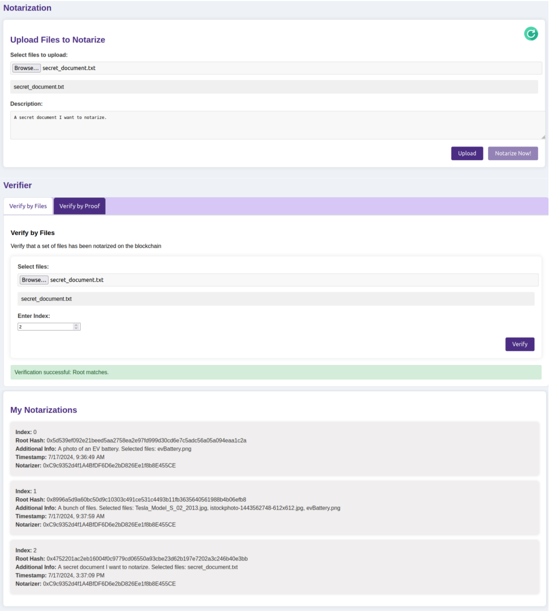
Regardless of the number of files to notarize, the cost of notarization remains constant, as only the Merkle tree root needs to be written to the blockchain. Because a Merkle tree is a complete binary tree data structure, building and verifying a Merkle tree on the client side remains efficient, requiring $\log n$ operations for verification, where n is the number of files in the tree. Figure 2 shows the demonstrator interfaces for all the described features.

The Notara smart contract is publicly accessible, allowing any user to store hashes on it. The smart contract may also record the user's address and the timestamp of the blockchain's block when the transaction occurs. An event can be emitted with this information, enabling the web application to easily retrieve notarized documents and their details by providing the file's hash or the notarization index. Events are indexed by bloom filters, facilitating efficient searches [12].

A key design decision is where the Notara smart contract should be deployed, which may depend on the use case's trust assumptions. For minimal trust, the Ethereum Mainnet is an option, albeit with higher operational costs (gas fees) and transaction delays. Conversely, if a permissioned network is trusted, then a solution based on a permissioned blockchain can be preferred due to lower operational costs. For example, for the notarization features of European public services such as the EU Digital Product Passport, the EBSI [1] (European Blockchain Service Infrastructure) is envisioned as a reliable permissioned blockchain platform for deploying a notarization smart contract.

V. Conclusion

This paper presents a decentralized blockchain-based notarization service that is fully customer deployable, as any user can have its own Notara instance by deploying the Notara smart contract in their preferred blockchain platform and configure the web application to interact with it. This eliminates the need for reliance on third-party infrastructure. Notara demonstrates the flexibility and security of deploying independent notarization systems tailored to specific organizational needs. By allowing users to manage their own notarization processes and only storing cryptographic proofs on the blockchain, Notara ensures data privacy, scalability, and cost efficiency. Our work emphasizes the potential of such customer-deployable solutions to enhance verifiability, integrity, and accountability across a wide range of IT applications. Our solution encourages further exploration of how customizable, decentralized notarization can transform data management and trust in digital ecosystems.



Notarization

Upload Files to Notarize

Select files to upload: secret_document.txt

secret_document.txt

Description:

A secret document I want to notarize.

Verifier

Verify by Files

Verify that a set of files has been notarized on the blockchain

Select files: secret_document.txt

secret_document.txt

Enter Index: 2

Verification successful: Root matches.

My Notarizations

Index: 0
Root Hash: 0x5d539ef092e21beed5aa2758ea2e97fd999d30cd6e7c5adc56a05a094eaa1c2a
Additional Info: A photo of an EV battery. Selected files: evBattery.png
Timestamp: 7/17/2024, 9:36:49 AM
Notarizer: 0xC9c9352d4f1A4BfDF6D6e2bD826Ee1f8b8E455CE

Index: 1
Root Hash: 0x8996a5d9a60bc50d9c10303c491ce531c4493b11fb3635640561988b4b06efb8
Additional Info: A bunch of files. Selected files: Tesla_Model_S_02_2013.jpg, istockphoto-1443562748-612x612.jpg, evBattery.png
Timestamp: 7/17/2024, 9:37:59 AM
Notarizer: 0xC9c9352d4f1A4BfDF6D6e2bD826Ee1f8b8E455CE

Index: 2
Root Hash: 0x4752201ac2eb16004f0c9779cd06550a93cbe23d62b197e7202a3c246b40e3bb
Additional Info: A secret document I want to notarize. Selected files: secret_document.txt
Timestamp: 7/17/2024, 3:37:09 PM
Notarizer: 0xC9c9352d4f1A4BfDF6D6e2bD826Ee1f8b8E455CE

Figure 2: Interfaces of the Notara demonstrator: the top interface allows users to notarize a set of files, the middle interface verifies if a file is effectively notarized, and the bottom one displays notarizations made by the connected address.

VI. Acknowledgments

Guillermo Toyos-Marfurt was supported by Uruguay's National Agency for Research and Innovation (ANII) under the code "POS_EXT_2021_1_171849".

References

[1] European Comission. The European Blockchain Services Infrastructure. https://ec.europa.eu/digital-building-blocks/sites/display/EBSI. Accessed: 2024-07-06.
[2] Council of the European Union. Directive 2009/125/ec, 2022. COM/2022/142 final.
[3] CSIRO. Decentralised Applications (DApps). https://research.csiro.au/blockchainpatterns/general-patterns/deployment-patterns/dapp/. Accessed: 2024-07-06.
[4] Omar Dib, Kei-Leo Brousmiche, Antoine Durand, Eric Thea, and Elyes Ben Hamida. Consortium blockchains: Overview, applications and challenges. *Int. J. Adv. Telecommun*, 11(1):51–64, 2018.
[5] J Michael Graglia and Christopher Mellon. Blockchain and property in 2018: At the end of the beginning. *Innovations: Technology, Governance, Globalization*, 12(1-2):90–116, 2018.
[6] Victoria Louise Lemieux. Trusting records: is blockchain technology the answer? *Records management journal*, 26(2):110–139, 2016.
[7] Commissariat à l'énergie atomique et aux énergies alternatives. Blockchain Verte. https://blockchain-verte.fr/. Accessed: 2024-07-06.
[8] Proof of Existence Project. Proof of Existence. https://https://proofofexistence.com. Accessed: 2024-07-06.
[9] Tonino Palmisano, Vito Nicola Convertini, Lucia Sarcinella, Luigia Gabriele, and Mariangela Bonifazi. Notarization and anti-plagiarism: A new blockchain approach. *Applied Sciences*, 12(1), 2022.
[10] Bitcoin Project. Bitcoin Core version 0.9.0 released. https://bitcoin.org/en/release/v0.9.0#how-to-upgrade. Accessed: 2024-07-06.
[11] OpenTimestamps Project. OpenTimestamps. https://opentimestamps.org. Accessed: 2024-07-06.

[12] Gavin Wood et al. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper*, 151(2014):1–32, 2014.

[13] Zibin Zheng, Shaoan Xie, Hongning Dai, Xiangping Chen, and Huaimin Wang. An overview of blockchain technology: Architecture, consensus, and future trends. In *2017 IEEE international congress on Big Data*, pages 557–564. IEEE, 2017.