On-Chain Open Notarization Protocol

Ivan Sala | slavni96@gmail.com

Abstract. This technical white paper explains some of the design decisions and functionalities of OSing contracts. It covers the contracts' features and the decentralized application functionalities. The aim is to provide a general overview over the notarization engine and possibility within the Ethereum Blockchain.

Keywords: Blockchain, Solidity, Open Source, Notarization, Signature, Ethereum

1 Introduction

OSign s a decentralized protocol build on top of the Ethereum Blockchain that helps you sign, notarize and verify any type of document and custom metadata. It gives you the ability to hash a document and upload it on-chain giving the possibility to be verified at any time.

2 Blockchain

A blockchain, is a growing list of records, called blocks, that are linked using cryptography. Each block contains a cryptographic hash of the previous block, a timestamp, and transaction data (generally represented as a Merkle tree).

By design, a blockchain is resistant to modification of its data. This is because once recorded, the data in any given block cannot be altered retroactively without alteration of all subsequent blocks. For use as a distributed ledger, a blockchain is typically managed by a peer-to-peer network collectively adhering to a protocol for inter-node communication and validating new blocks. Although blockchain records are not unalterable, blockchains may be considered secure by design and exemplify a distributed computing system with high Byzantine fault tolerance. The blockchain has been described as "an open, distributed ledger that can record transactions between two parties efficiently and in a verifiable and permanent way".

3 Notarization

Notarization is the official fraud-deterrent process that assures the parties of a transaction that a document is authentic, and can be trusted. It is a three-part process, performed by a Notary Public, that includes of vetting, certifying and record-keeping. Notarizations are sometimes referred to as "notarial acts." The part of the Notary Public could, given the design of the Blockchain, be taken from this new entity. This breaking change could create endless possibility in the notarization world.

4 Contracts

Blockchain-based smart contracts are proposed contracts that can be partially or fully executed or enforced without human interaction.[56] One of the main objectives of a smart contract is automated escrow. A key feature of smart contracts is that they do not need a trusted third party (such as a trustee) to act as an intermediary between contracting entities -the blockchain network executes the contract on its own. This may reduce friction between entities when transferring value and could subsequently open the door to a higher level of transaction automation

4.1 Solidity

Solidity is an object-oriented, high-level language for implementing smart contracts. Smart contracts are programs which govern the behaviour of accounts within the Ethereum state.

4.2 Code

Below the code of the main entity used in the process of the notarization. This contract have been written keeping in mind to be as simple as possible. Users will be able to notarize on the blockchain a entity called "Document" that will contain:

- The address of the owner
- The creation timestamp
- Custom Metadata to better define the document context

All this information will be stored inside an array called "_documents".
Two methods will be provided to manipulate the chain:

– sendDocument
– getDocument

The first method "sendDocument" will let users write the MD5 Hashed document with custom metadata inside the blockchain. The second will let everyone able to interrogate the chain to discover the entity given the MD5 Hash of the uploaded document.

This workflow can work as a third-party that assures the transaction of a document and can be trusted beyond any doubt.

DocumentUploader

```
// SPDX-License-Identifier: MIT

pragma solidity 0.8.0;
pragma abicoder v2;

contract Signer {

    mapping(string => Document) private _documents;

    struct Document {
        address owner;
        uint256 timestamp;
        bytes metadata;
    }

    function sendDocument(string memory documentHash,
                          uint256 timestamp,
                          bytes memory metadata) public {
        _documents[documentHash].owner = msg.sender;
        _documents[documentHash].timestamp = timestamp;
        _documents[documentHash].metadata = metadata;
    }

    function getDocument(string memory documentHash)
                          public
                          view
                          returns(uint256 timestamp,
                          bytes memory metadata) {
        timestamp = _documents[documentHash].timestamp;
        metadata = _documents[documentHash].metadata;
    }
}
```

5 References

1) What is blockchain URL: https://en.wikipedia.org/wiki/Blockchain
2) What is notarization URL: https://www.nationalnotary.org/knowledge-center/about-notaries/what-is-notarization
3) What is Solidity URL: https://docs.soliditylang.org/

6 Disclamer

This paper is for general information purposes only. It should not be relied upon for accounting, legal or tax or recommendations.