A Blockchain Self-Sovereign Solution for Secure Generation, Exchange and Management of User Identity Data

Nicolae Ghibu[1], Augustin Jianu[1], Alexandru Lupascu[1] and Ştefan Popescu[2]
[1]*Certsign, Romania*
[2]*University of Bucharest, Romania*

Keywords: Blockchain, Self-Sovereign Identity Scheme, Authentication of Identity.

Abstract: In this day and age the need for a secure and easy to use communication system between individuals and service providers has grown past the point of it being a matter of comfort and has entered the field of necessity. The traditional paradigm for such a system used to be a centralized digital model, where the platform owner was the main trusted 3rd party and maintained a log for each individual. This manner of providing and exchanging identity data, through centralized digital entities, is inefficient in terms of duplication, at best, and it presents considerable risks for the users, at worst. The proposed system aims to be a practical solution to these problems, being a self-sovereign identity scheme that can be used by governments, businesses, and users that need to make a claim verification alike.

1 INTRODUCTION

Digital identity authentication guarantees that individuals are who they claim to be or that they have the necessary credentials to access a service. Subject authentication, along with protecting key information, are two main objectives in identity management. To avoid stealing and manipulating credentials or having to reveal unnecessary information, some service providers implement solutions that require multi-factor authentication (Nagaraju and Parthiban, 2016), which further complicates the system. This type of approach also allows the control and ownership of data to remain in the hands of the service provider.

One alternative to having service providers entrusted with the storage and management of identity data is for the owner of the actual data to be entrusted with storing and managing their own credentials using a personal or company device and only revealing the necessary information required for a given direct interaction with a service provider. This model is called "Self-Sovereign" identity.

One of the main difficulties that needs to be addressed is the absence of any standard way to verify digital credentials. Trusting third parties with your data carries the risk of misuse or susceptibility to integrity threats. The blockchain is becoming the most often used alternative in domains such as IoT security, healthcare, business, and many others, where users do not trust third parties and are aware of data collection and its usage (Syed et al., 2019). Studies (Sharma et al., 2020; Leka et al., 2019) have shown that smart contracts over the blockchain have become increasingly used in areas such as healthcare and education.

In their white paper (Sovrin Foundation, 2018), the Sovrin Foundation proposed "A Protocol and Token for Self Sovereign Identity and Decentralized Trust", an open-source blockchain protocol with an accompanying token that tackles the problem of sovereign identity. They do this by firstly creating a standard format for digital credentials that can easily be automated, and secondly by creating a standard way to verify the source and integrity of said digital credentials. For the second objective, they propose a private/public key pair system - the first used for signing the document and the second used for verifying the signature. A blockchain, whether public or private, seems to offer some of the desired advantages, like being a decentralized root of trust that is not owned by anyone but can be used by everyone or every registered user, and thus seems ideal for serving as a decentralized self-service registry for public keys.

Some (Lupascu et al., 2021) have identified certain problems with the specific implementation proposed by the Sovrin Foundation. Namely, their design as a "public global utility" requires the usage

Ghibu, N., Jianu, A., Lupascu, A. and Popescu, Åd.
A Blockchain Self-Sovereign Solution for Secure Generation, Exchange and Management of User Identity Data.
DOI: 10.5220/0012148800003538
In *Proceedings of the 18th International Conference on Software Technologies (ICSOFT 2023)*, pages 744-750
ISBN: 978-989-758-665-1; ISSN: 2184-2833

of their Sovrin Blockchain, which has been designed only for identity claims and for their own Sovrin Token. This requires any organization that would like to implement this solution to alter its business model for compatibility reasons, such as accommodating crypto assets like the Sovrin Token. Another identified problem is that an identity claimed verified by a third party cannot be unilaterally revoked before the expiration date by said third party. For example, a driver's license or student pass cannot be revoked by the state issuer or by the university, respectively.

In this paper, based on certSIGN's patent application (Lupascu et al., 2021), we present a novel and flexible solution to the self-sovereign identity problem that not only addresses the core challenges but also introduces the possibility of implementing smart contracts between users. Our approach builds upon existing an existing implementation and leverages blockchain technology to create a secure and decentralized framework for identity management. The project is currently in late stage development.

In the following sections, we will begin by providing essential background information about the proposed solution, outlining its specific benefits, and offering practical examples to illustrate its functionality. We will present an in-depth overview of our proposed approach, starting with a basic use-case scenario and addressing the potential challenges that can arise and will delve into the roles and responsibilities of each participant within the network, highlighting their interactions. We will provide a chronological list of data transactions within a generalized scenario, illustrating the sequence of events and data flow among the participants. For better clarity, we will showcase a step-by-step interaction between the different entities and how they engage with the system. Finally, we will draw conclusions based on our findings and discuss potential avenues for future work.

2 THE PROPOSED SOLUTION

2.1 Overview

The solution (Lupascu et al., 2021) claims to meet the self-sovereign identity requirements in a more business-friendly way. Specifically, it can be implemented over any type of blockchain, whether public or private, and does not require the acquisition of any specific crypto asset. While the proposed solution refers to Bitcoin as the blockchain in use for convenience purposes, it can be applied to any generic blockchain. Because blockchains are immutable, storing personal data on a blockchain is strongly discouraged, even if said data is in encrypted form. People can reveal their private keys by mistake, and digital information can get leaked, so good practice dictates that all the information stored on the blockchain has to be hashed, and only the hashes are stored on the blockchain. It is also assumed that blockchain users carry digital identifiers that are used to identify them by way of digital signatures that accompany each transaction they take part in. User identity information can be hashed and then encrypted to be stored on the blockchain at the time of enrolling the user on the platform. Management trust is provided by a root user/ platform owner that authorizes new users and user roles, such a the role of a trusted third-party, by recording their authorization on the blockchain.

Users have the ability to create their own digital identifiers and trusted third-party users can enroll identity owners (users who will use the platform for their ID authentication) by collecting their data in a secure way such as face-to-face verification, and storing encrypted hashes of the data on the blockchain alongside the user's digital identifier. Each data exchange between users must be authorized by the trusted third-party that verified the user's attributes. This authorization process involves a multi-stage verification protocol that accesses information stored on the blockchain. These interactions between users, or between users and trusted third-parties are also archived on the blockchain.

By utilizing information stored on the blockchain and a user identifier (ex: the user's public key) provided by the user, a computing device can obtain a particular assertion from the multiple proofs recorded on the blockchain. The assertion obtained by the computing device can be checked by using information provided by the user, alongside a secret key, via the blockchain, from a trusted third-party. The secret key is generated based on two public keys: one associated with the computing device that wants to check the assertion and the other based on the computing device controlled by the trusted third-party. This approach allows users to choose which aspects of their identity they want to share with information consumers.

For example, a student accessing a virtual library might want to share a proof that containing their name and university status, but not their home address. In this scenario the library acts as the information consumer, the student as the user, and the University that emitted the student license as the trusted third-party. The library wants to check the assertion that the user is a student, and based on the user's identifier, information stored on the blockchain, and after a exchange of information with the University via the blockchain with the aid of a secure communication protocol it can

confirm, with a high degree of assurance, that the user is actually a student or not, without learning any other information about them.

One of the specific benefits of the solution, described later on in detail, is its independence of the type of blockchain used and that it provides methods by which a trusted third-party, that has verified the identity claim of a user in the past, can revoke any assertion that is based upon said past verification.

The system architecture is specifically designed to address the self-sovereign identity problem where a reference dataset is associated with a user's identifier (e.g., public key) and two computing devices are involved. The first device stores the identifier along with a high assurance level that a first dataset accurately represents the reference dataset, without storing the first dataset itself. The second device stores the identifier with a second dataset that represents a subset of the reference dataset. Without disclosing the second dataset to the first computing device, the second device needs to verify, with a high degree of certainty, that the first dataset accurately represents a subset of the reference dataset.

2.2 A Basic Use-Case Scenario and the Problems that Can Appear

We present the following working scenario: An identity owner needs to provide specific data attributes to an online service provider, who acts as a a data consumer, in order to be provided a particular service.. The data attributes can be static, such as a social number, or dynamic, such as a residency, or the place of work. When the data consumer requests the authentication of a set of attributes claimed by the identity owner, the request is recorded on the blockchain and then answered by one or more special attributed "trusted third-party" users. This request is also associated with identifiers for both the identity owner and the trusted third-party, permitting the option of payment between the two for each such request. The request transaction can include additional information that can be used by the trusted third-party to create a secure mode of communication.

The role of the trusted third-party is essential because without them, data consumers would not be able to verify, with a high degree of accuracy, that the data attributes presented to them are authentic. On the other hand, identity owners couldn't easily share just the parts of their data attribute that are relevant to the service they want to access, without revealing a multitude of other information. The solution presented can overcome such challenges by allowing the identity owner to encrypt only specific data attributes to be shared with a data consumer, providing the data consumer with a secret key via a secure communication protocol alongside the encrypted data attributes that are needed for the service. The secret key can be encrypted with the data consumer's public key. Furthermore, the blockchain can be an ideal system for sharing these encrypted attributes and for the data consumer to receive information on the assurance level of said data attributes. The trusted third-party also gets to share said information to the data consumer without having to store copies of the data attributes in question.

2.3 The Composition of the Computing System

The computing system comprises a network of interconnected computing nodes that are associated with at most one platform owner, at least one trusted third-party, and a various number of data consumers and identity owners. The platform owner has the initial role of authorizing network participants to verify identities, giving them the status of a trusted third-party, and defining the types of data attributes that are verified, shared, and validated. For example, the platform owner can authorize a network participant, such as a university, to verify identities for their students and define such data attributes as a string representing a student ID number, a date representing the year the student enrolled, etc. But the platform owner might not define information such a the home address of the student.

The trusted third-parties are the network participants that verify the identity of other network participants with a high level of assurance.

The identity owners in the network are the participants that make their identity data attributes available for distribution over the network.

The data consumers in the network refer to the participants that receive data attributes over the network and need to check their validity. Therefore, they request and receive information regarding the assurance levels of the identity assertions received over the network. Depending on the implementation, a user can have a single role in the network or multiple roles. For example, a participant that has authorized other participants as trusted third-parties might himself act as a trusted third-party in certain situations, or even an identity owner.

The blockchain ledger is composed of records of all the transactions related to the digital identities of the participants, and the data attributes that have been verified and distributed. These transactions are recorded in chronological order and include

authorizations granted by the platform owner, attribute verification proofs submitted by trusted third-parties, and authentication requests submitted by service providers which act as data consumers.

In certain implementations, the ledger may also contain executable programs known as ”smart contracts.” These programs automate several aspects of data sharing transactions, and they can be used to facilitate the performance of tasks such as identity and data validation.

Overall, the blockchain ledger serves as a repository of all activity within the network which is both secure and transparent It ensures that all participants have access to the same information, and that all transactions are executed in a trustworthy and accountable manner.

2.4 Chronology of the Data Transactions Being Submitted to the Ledger

Transactions with a ledger always include at least the public identifier of the issuer, alongside a digital signature of the owner of said public identifier. An illustration regarding the following steps can be seen in Figure 1.

The first transaction on the ledger is performed by the platform owner, in which they define what identity data attributes are acceptable on the platform. This should include a data payload consisting of data attribute definitions that can also serve as reference points in the ledger for other transactions made by participants.

Another type of transaction performed by the platform owner is to submit a data entry, designating other participants as an authorized trusted third-parties. These transactions may also include the public identifier of the trusted participant, thus informing other participants that the designated participants can now act as a trusted third-parties.

A trusted third-party can submit a transaction that stores on the ledger proofs of data attributes associated with the public identifier of an identity owner. The data attributes in question must be of the types defined by the platform owner.

A data consumer can receive one or more data attributes from an identity owner and then issue an authentication request by submitting a transaction on the ledger. To initiate such a request, the data consumer needs firstly identify the transaction on the ledger where a trusted third-party stored data attribute verification proofs regarding the identity owner in question, and then direct that request to the specific third-party. Another approach would be associating the identity owner’s public identifier with a ”smart contract” so that only the issuer of data attribute (a specific trusted third-party) is allowed to respond to the the data consumer’s request. These types of transactions may also include payments from the data consumer to the trusted third-party.

The trusted third-party can respond to the request submitted by the data consumer with their own transaction on the ledger. The response does not reveal any of the data attributes of the identity owner but can be used by the data consumer to prove, with a high level of assurance, the authenticity of said data attributes.

Moreover, as mentioned before, the trusted third-party can revoke previous transactions that verify the data attributes of an identity holder. Verification proofs can also be revoked, which is useful in instances where one or more data attribute definitions become deprecated or are totally revoked.

2.5 Step-by-Step Illustration of Interactions Between the Devices Controlled by Network Participants

In this section, we will describe briefly how each participant interacts with each other, starting with the attribution of roles, the definition of data attributes, and the verification of a particular data claim made by an identity owner. An illustration of the steps is presented in Figure 2.

The first stage is when the platform owner defines the particular data attributes that can be made available for distribution. The definition of a data attribute should include at least an identifier and one data type. The computing device associated with the platform owner records these definitions on the ledger as a first transaction, which can be used as a reference point for future transactions.

In the next stage, the platform owner enrolls a trusted third-party in the network, enabling them to verify, with a high degree of certainty, authentication claims and record verification proofs. At this point the platform owner records a transaction on the ledger, making all other participants aware of the newly authorized trusted third-party. This may sometimes involve recording a public identifier of the trusted third-party. In other cases, the enrollment may require recording several public identifiers, which may all be authorized to act on behalf of the newly appointed trusted third-party.

At the third stage, an identity owner uses their computing device to generate a digital identifier that will be used throughout the network. In some implementations, this identifier can be a public-private key pair that will be further used to encrypt, decrypt, and

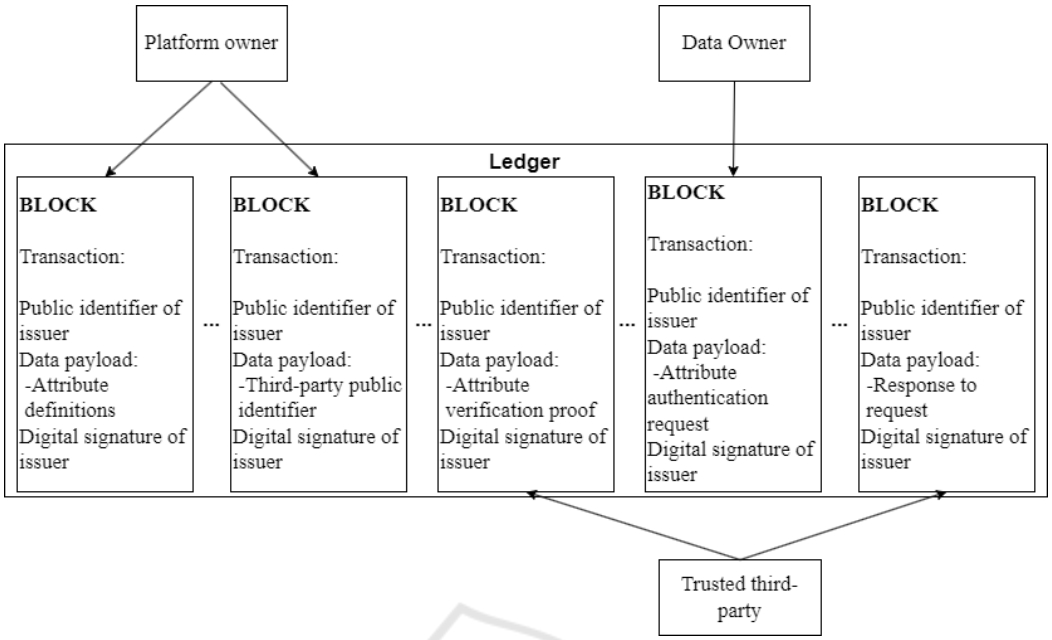

Figure 1: Generic transactions stored on the ledger by network participants.

sign data.

At the fourth stage, the identity owner presents evidence of their data attributes to the trusted third-party for verification. This step requires a physical interaction between the two involved parties. The identity owner uses their computing device and the identifier generated in the previous stage to create a signed consent for data attribute verification and data processing, and sends it to the trusted third-party.

At the fifth stage, the trusted third-party generates the resulting verification proofs and stores them on the ledger, alongside the identity owner's consent. Some variations of this step involve hashing each individual data attribute and then encrypting them before storing them on the ledger. The encryption is done by generating a secret key that the trusted third-party stores in a tamper-resistant key manager on their computing system. This same key will be used later on, at stage nine, to decrypt the data attribute hashes.

In stage six, the identity owner's computing device generates and signs an identity claim that includes their public identifier generated in stage three, as well as one or more attributes required to access a service. This type of interaction is advantageous in scenarios where the data owner does not wish to disclose all of their attributes, but only the necessary ones.

In stage seven, the identity owner sends the generated identity claim to a data consumer who is responsible for identifying users requesting a particular service. This stage usually involves establishing a shared secret over an insecure channel.

In stage eight, the data consumer's computer device generates an identity request that is stored on the ledger. The request consists of the identity owner's previously obtained public identifier and the identifiers of the claimed attributes. In this type of protocol, the data consumer does not disclose to the trusted third-party that will be charged with verifying the identity claims the values of the attributes received by the identity owner. The authentication requests also includes information necessary to establish a secure channel of communication between the two.

In stage nine, the computation device of a trusted third-party responds to the previously generated request. The computation device checks the attribute identifiers included in the request and extracts only specific verification proofs out of all the ones generated at stage five. If the request includes identifiers for which there is no verification proof, or that said proofs have been revoked, then the request is met with a failure to authenticate. Otherwise, the trusted third-party confirms the attributes of the identity owner using a secure communication protocol with the data consumer.

At the tenth stage, the trusted third-party records the verification result on the ledger. This is necessary in cases where there needs to be an unalterable record of access to identity data without recording the actual identity data or anything that might be used to reconstruct it.

In stage eleven, the data consumer receives the re-

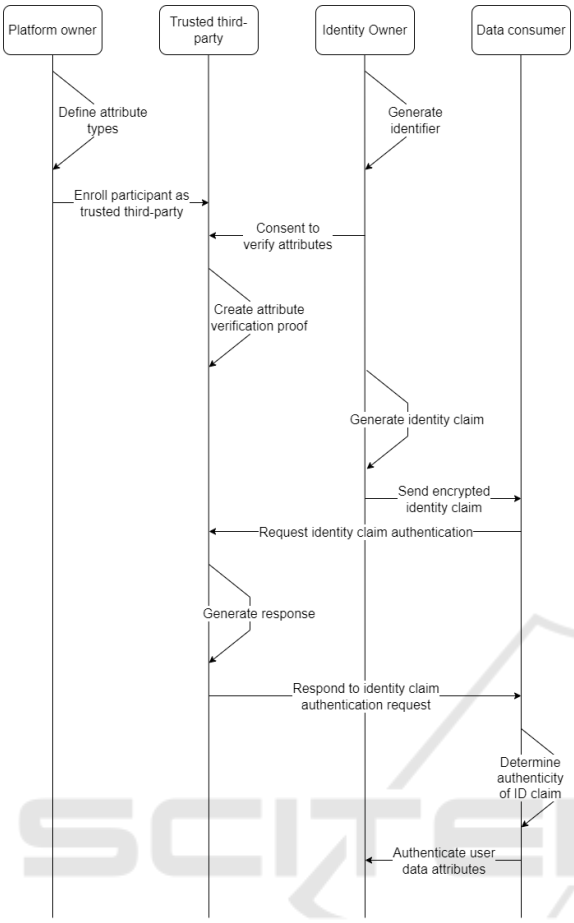

Figure 2: Step-by-step interaction of platform participants.

sult of the previous stage and determines the authenticity of the identity claim received at stage seven. The computing device of the data consumer reads the response recorded on the ledger in the previous stage and uses that information to build a secure communication protocol with the trusted third-party. Over this secure communication system, the identity claim is confirmed by the trusted third-party via the reconstruction of a message authentication code recorded on the ledger in encrypted form at the previous stage. This protocol is useful because, in this way, the data consumer is left with unalterable proof that the trusted third-party actually confirmed the authenticity of the identity owner's claims.

At the last stage, the computing device of the data consumer triggers the sending of a message to the identity holder, informing them of the results. This technique is useful in the scenario where the identity owner needs to receive a service over an untrusted channel, but the data consumer needs to trust the data provided by the identity owner.

3 CONCLUSIONS

The solution described in this paper offers a somewhat original approach to addressing the self-sovereign identity problem. It provides a flexible and secure solution for communication between all network participants via a blockchain alongside a public/private key communication protocol. This allows for a more secure information exchange between network users, which is important when dealing with identity attributes.

A main advantage of their proposed solution is the flexibility it grants, as it is independent of any specific blockchain or crypto asset. This means that it can be adapted to work with any blockchain, thus being a significant advantage for companies and organizations that already have a specific blockchain in place.

The solution is also able to offer a payment mechanism for each proof generated, which provides incentive for users to participate in the network and generate proofs only when necessary. This is important because it helps ensure that relevant attributes are the ones being checked. This can improve efficiency and reduce error risk.

Most importantly, the solution is designed to work under a zero-knowledge framework, which means that it does not require the disclosure of any unnecessary information, thus ensuring the privacy and confidentiality of the identity users.

The use of a blockchain framework also provides significant benefits, namely it allows for the secure distribution of data attributes among independent users and service providers. This helps ensure security, transparency, and access right management, which are all critical factors when dealing with sensitive information.

Another notable advantage of the proposed solution is its ability to record binary files on the ledger, which can act as "smart contracts" between network participants. This allows for some more complex interactions between network users, which greatly increases the practical scenarios where it can be utilized.

Overall, the solution proposed offers a flexible and secure approach to addressing the self-sovereign identity problem, with the potential to provide significant benefits to organizations across a range of industries. The main challenge with this approach lies in developing a governance framework that reaches a balance between decentralized control and centralized oversight, fulfilling the needs and interests of the participants while ensuring compliance with legal and ethical standards.

Regarding future work, the proposed solution is

currently in its developmental stage and will undergo various testing and refinement as it reaches its beta stage. Continuous collaboration with the development team will be crucial for further enhancements and improvements.

ACKNOWLEDGEMENTS

This research was supported by the European Regional Development Fund, Competitiveness Operational Program 2014-2020 through project IDBC (code SMIS 2014+: 121512)

REFERENCES

Leka, E., Selimi, B., and Lamani, L. (2019). Systematic literature review of blockchain applications: Smart contracts. pages 1–3.

Lupascu, A., Jianu, A., and Ghibu, N. (September 2021). System and method for secure generation, exchange and management of a user identity data using a blockchain, european patent application number 20164682.5.

Nagaraju, S. and Parthiban, L. (2016). Secauthn: provably secure multi-factor authentication for the cloud computing systems. *Indian Journal of Science and Technology*, 9(9):1–18.

Sharma, A., Tomar, R., Chilamkurti, N., and Kim, B.-G. (2020). Blockchain based smart contracts for internet of medical things in e-healthcare. *Electronics*, 9(10):1609.

Sovrin Foundation (January 2018). A protocol and token for self-sovereign identity and decentralized trust.

Syed, T. A., Alzahrani, A., Jan, S., Siddiqui, M. S., Nadeem, A., and Alghamdi, T. (2019). A comparative analysis of blockchain architecture and its applications: Problems and recommendations. *IEEE access*, 7:176838–176869.