Volume 11, Issue 3
June 2022

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

A Blockchain-based Conceptual Framework for Privacy Preserving Self-Sovereign Identity with Selective Disclosure

Jayana Kaneriya and Hiren Patel

[1] Sarva Vidyalaya Kelavni Mandal, Gujarat, India

E-mail: jayana102006@gmail.com

ABSTRACT

Identity cards have undergone a significant change from simple paper documents designed for single application to smart electronic cards. Various countries use biometric cards, electronic national cards, portable Ids or mobile Ids for identification of an individual to offer various services, particularly in India, different ID cards are utilized for different purposes (E.g., Driving Licence, Aadhar Card, Pan Card, Employee ID Card, etc). All these cards are issued by centralized identity-based frameworks, which give binary identification of an entity, which implies either the entire identity attributes are known or nothing is known. Also, it experiences lack of transparency, identity owner controllability, traceability, security and privacy. With the advancement of blockchain technology, new decentralized and transparent digital identity models are proposed to address these challenges. Self-sovereign identity is the latest evolution in the identity domain which provides complete ownership of data to the identity owner with privacy. Our aim is to explore the potential of blockchain technology for use in the management of self-sovereign identity, where numerous researchers and academicians are keen on setting up an open and decentralized environment. We further investigated the work done by existing specialists in the said domain and based on that we proposed a new blockchain based self-sovereign identity model with smart contract. Our model empowers users (employees, students, customers, etc.) to share information to service providers (employers, universities, banks, etc.) on a need-to-know basis and also allows them to validate that information.

Keywords: *Blockchain technology, self-sovereign identity, smart contract*

1. INTRODUCTION

Rapid advancement in technologies and smart applications used in various domains empower the applied vision of smart city. By 2025, it is expected there will be in excess of 75 billion smart devices connected via private or public networks [1]. Although these smart devices have brought tangible benefits to modern society, privacy issues have become worthy of attention in Smart Cities. Personal identity data of users are sensed, collected, stored, spread anytime, and can be further analyzed by service providers and third parties, which can lead to privacy and security issues. To address such issues, numerous identity management frameworks have emerged which can satisfy the growing demands of online services. Yet, the majorities of these are centralized and rely on password, multi factor authentication, tokens, voice assistance, fingerprint, facial recognition, and smart cards which are vulnerable to various threats and impose challenges [2]. In such systems, service providers can get advantage from identity owners on the Internet. The owner of the identity loses control of it under such non-transparent systems [3]. In addition, biometric identification systems suffer from numerous problems like security and privacy [4]. Equifax security break in 2017 compromises around 145 million individual's identity information [5] and also, Facebook gives unapproved admittance to Cambridge Analytica around 87 million user's personal information [6]. Due to this, data protection laws in the European Union are included in the new GDPR, and federated privacy laws in the U.S., CCPA, are the most significant advances in privacy developments [7]. Blockchain powered identity frameworks with provable credentials have the capability of turning into the establishment of easily accessible, secure and private identities in which untrusted parties are involved [8]. Through this research we aim to design a framework which provides identity owners to selectively disclose identity attributes with security and privacy.

There are various application areas in which identification frameworks assume a significant importance, for example, healthcare services where in patients' clinical and identification information must be controlled and accessed by themselves alone and educational and professional data of doctors and other staff needs to be private and protected [9]. So as to demonstrate the identity of people in various places, for example to visit outside nations, visas and other government endorsed identification certificates must be secured [10]. Customers may apply for loan or share economic transactions without any intervention from third parties and thus the bank can provide their customers with a better KYC (Know Your Customer) service [11]. In the education sector, students can share their educational certificates as a proof with employers or other institutions with consent and control [12]. Moreover, Blockchain-based land registry systems are analyzed in [13].

As academic researchers, we conclude that blockchain based self-sovereign identity can help owners to share their identity attributes in day-to-day life (i.e., loan approval, medical insurance, job, vote, discount, travel pass, and drive a vehicle etc.) without losing privacy. Such systems protect credentials from fraud and misuse. Henceforth, the application space of the identification framework is extremely wide and along these lines, researchers have created decentralized identity models that can comprehensively be grouped among public, private and

hybrid. Each model has its own advantages and challenges that we will address in the next section.

Through this research, we wish to address the following research questions: How computational advancements, for example distributed ledger technology (Blockchain), can be used to create self-sovereign identities for the identity owners with the goal that ownership, privacy, and security be improved? To be specific, (a) Credentials should be issued in such a way that it should be verified without any central control. (b) Credential owner is able to share only required attributes to verifier (c) identity owner is able to share proof of specific identity attributes (d) without consent of identity owner, any other entity should not be able to use or verify (e) privacy and security of credentials should be maintained with or without trusted verifier. We propose a conceptual framework to selectively share attributes using blockchain technology. Our model uses Blockchain with smart contracts to automate every process.

The rest of this paper is organized as follows. In Section 2, we introduce related works on blockchain based identity models with research challenges and opportunities. Then, Section 3 describes our proposed method used in the framework for credential issuance. In section 4, we present our proposed conceptual framework with components. Section 5 discusses tools used to set up a decentralized environment with results. Finally, in section 6, by outlining the potential future direction of the identification scheme, we conclude our paper.

2. RELATED WORK

With the advancement in Blockchain technology, decentralized identity management has gained lots of interest in the last decade. In this regard, the W3C community has proposed standards for decentralized identifiers (DID) [14] and verifiable credentials (VC) [15]. DIDs are designed with goals such as decentralization, control, security, portability etc. and verifiable credentials are declaration of temper proof and privacy preserving statements about an entity.

Sovrin foundation, a non-profitable community, has introduced the Sovrin ledger which can record claim definitions without any central control [16]. A zero-knowledge proof based Sovrin protocol enables claim privacy with selective disclosure. In the Sovrin network, steward nodes are responsible for validation of transactions which uses PFBT consensus protocol which negates decentralization [2]. An Uport is a smart contract based secure digital identity solution which enables users to create multiple digital identities. To recover Uport identity, it uses a recovery mechanism with recovery delegates like family, friends, organizations which can impose security issues in Uport [17]. Public Ethereum based LifeID and selfKey solutions support secure key recovery mechanisms [18, 19] while a private Ethereum based EverID enables users to share credentials with consent [20]. Sora is built on permissioned blockchain, Hyperledger Iroha, with mobile application to share selected attributes to the verifying authority [21].

Other works focused on new types of decentralized authentication systems which combines Blockchain with biometric user-controlled authentication schemes such as the Horcrux protocol [22] and other research based on Cancellable Biometrics with verifiable claims enable users with privacy preserving identities which guarantee single time enrolment of a user in the system [23].

Mustafa [24] introduce SCPKI, a smart contract based transparent web of trust model to verify identity attributes of an entity. Due to cost associated with Blockchain storage which degrades throughput of the system, proposed model uses IPFS to store large amount of data and to maintain integrity of the data, hash of data is stored on Blockchain. Moreover, he also implemented model with two versions, full version and light version. The performance of the light version is high as it does not store attributes within contract and so gas cost is low but full version of contract can access attributes of other contracts and so gas cost is high. Similarly, Benedict Faber et.al [3] recommended BPDIMS, which is a conceptual GDPR compliant transparent human centric model. The researchers have highlighted concerns regarding user consent for data processing, storing and selling and data monetization. However, this work does not cover the implementation details with results.

Jamila et al. [25] present a DNS based novel model for Blockchain based identity management with security and privacy. The evaluation result shows that the proposed model is highly scalable with a large number of validators and subsequently provides future research directions. Along with credential verification, a credential issuance is also an important procedure in identity management where a single trusted authority performs issuance of a credential. But the scheme proposed by Alberto Sonnino et al. [26] is based on distributed ledger and threshold issuance which provides selective disclosure while preserving integrity of credential attributes. They further explore possible use cases with the implementation library in chainspace and Ethereum. Harry Halpin [27] proposes a Nym credential which is a cryptographic credential with unlinkability with the transactions for outsider entities.

Jeonghyuk Lee [28] discusses the application of zero-knowledge proof in identity domain covering commit and prove method to address privacy issues. They further explore possible applications of the model for minor check, address check, transcript check and work history check. The scheme designed by Kwame Omono Asamoah et.al [29] discusses reliable identification of a particular citizen in smart city applications using group signature and zero-knowledge proof. Xiaohui Yang et al. [30] propose a new claim identity model, which aims to provide unlinkability between attributes with its owner by creating privacy tokens. They designed smart contracts to automate attribute proof generation and verification using zero-knowledge proof.

Volume 11, Issue 3
June 2022

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

3. PROPOSED METHOD

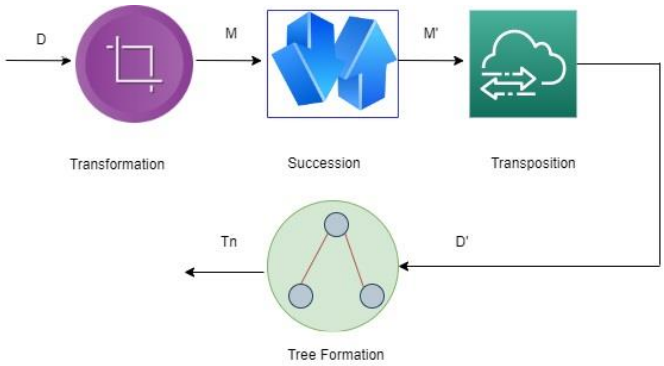

Fig. 1 Process of Proposed Method

Algorithm 1 : calculateMole()

Inputs: attribute
Results: mole

```
Algorithm calculateMole Starts
    assign zero to mole
        for each character in attribute do
            mole <- mole + ASCII (character)
        end for
    return mole
Algorithm calculateMole Ends
```

Algorithm 2 : hashTree_Formation()

Inputs: attributeList
Results: hashTree

```
Algorithm hashTree_Formation starts
    for each attribute in attributeList do
        hashTree.push(hash(attribute[i]))
    end for
    len=length(attributeList)
    offset = 0;
    repeat
        for i = 0 to  len – 1 do
            hashTree.push(
            hash(hashTree[offset + i],
                hashTree[offset + i + 1]))
            i=i+2
        end for
    offset = offset+len
    len= len/ 2
    until len>0
Algorithm hashTree_Formation ends
```

Algorithm 3 : proposed_Method()

Inputs: attributeList
Results: hashTree

```
Algorithm proposed_Method starts
    for each attribute in attributeList do
        Mole[i] =caclulateMole(attribute[i])
    end for
    OrderedMole=QuickSort(Mole)
    UpdatedAttributeList=Tranpose(Mole,OrderedMole)
    hashtree=hashTree_Formation(UpdatedAttributeList)
Algorithm proposed_Method ends
```

Using the proposed method, the prover can convince the verifier that a specific attribute belongs to a credential without disclosing other attributes. The most notable feature of the proposed mechanism is the use of a pre-processing layer on top of an n-ary tree construction, which allows for randomization of credential attributes. Moreover, the proposed structure can reduce proof size with the increase of attributes that leads to minimization in information leakage and prevention of various active and passive attacks. Therefore, to enhance security and reliability of the tree based selective attribute proof verification, the proposed method uses two layers, one is pre-processing and other is tree construction. Figure 1 shows processes of proposed method and algorithm 1, 2 and 3 shows details about proposed method. The following section discusses these layers in detail. To provide random attribute selection with equal probability, the pre-processing step comprises three main operations: transformation, succession and transposition. The transformation, as credential attribute values come in mixed formats like numeric, alphanumeric, image etc., converts attribute value into numeric value for further processing. So, for each attribute, mole value is calculated by adopting ASCII value, Hash value, linear equation, quadratic equation, exponential equation and rational equation using formulas listed in Table 1 and Table 2 displays used notations. Succession operation rearranges order of attribute's mole values using ascending or descending order and finally, transposition operation changes the sequence of the attribute set.

Table 1
Mathematical formula for Mole value (m) calculation

Type	Mathematical formula
ASCII Value	$m = \sum_{i=1}^{l} (D_{c,j})$
Hash Value	m=X=Hash ($D_{C,j}$) % p
linear equation	m=(a*X+b) % p
quadratic equation	m=(a*(X^2)+(b*X)+c)% p
exponential equation	m=(a*(X^e)+(b*X) +c)% p
rational equation	m=((X/a) +(b*X) +c) % p

Volume 11, Issue 3
June 2022

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

Table 2
Summary of notations

Notation	Meaning	Notation	Meaning
E	Entity	T_n	n- ary tree
E_{CIS}	Credential Issuer Entity	RH	root hash
E_{CV}	Credential Verifier Entity	CT	Consent Token
E_{IO}	Identity Owner Entity	P	Proof
C	Credential	E	Exponent Value
D	Attribute Set	a, b, c	Constant values
$D_{C,j}$	j^{th} attribute of credential	k	Level of the tree

In tree construction layer, updated attributes are set as an input to n-ary bottom-up tree construction operation. If the numbers of attributes are less than n^k, then we place the last attribute value in all remaining positions.

Then, n values are combined together to generate intermediate nodes and finally, using SHA256 hash function, we generate a hash root node.

4. PROPOSED FRAMEWORK

4.1 OVERVIEW

Identity services tend to operate in an untrusted environment and it involves various entities to share and verify digital credentials (C). There are three main entities (E) in our framework, namely Credential Issuer, Identity owner and Credential Verifier. Credential issuer is responsible for issuance or revocation of valid credentials to identity owner and stores it on blockchain.

Identity owner possesses various credentials and is able to share credential attributes with consent to service provider or verifier. Credential verifier is responsible to validate attributes provided by the identity owner. The overview of the framework is illustrated in Figure 2.

Our conceptual framework consists of software components like core services, application programming interfaces (API) and smart contracts. Core services are responsible to provide smooth and secure interaction between entities and data storage.API consists of functions which allow interaction with different software components.

Smart contracts maintain hashes of identity credentials for integrity and entities can share consent to provide limited access for verification. In the following sections we outline specifically the role of said software components.

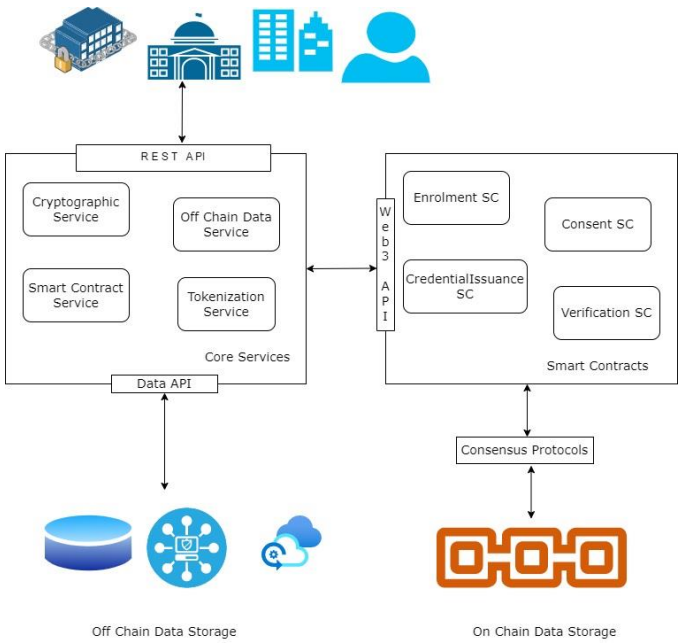

Fig. 2 SSI Framework using selective disclosure

4.2 SOFTWARE COMPONENTS

Our conceptual framework consists of three main software components: Core Services,Application Programming Interfaces and Smart Contracts.

Core Services: Core services include services like cryptographic service, on chain and off chain data storage service, smart contract service and tokenization service. Smart contract service helps entities and other services to interact with Blockchain deployed smart contracts. Cryptographic service is responsible for all cryptographic operations (key generation, key management, encryption, decryption, digital signature) within the system. Symmetric encryption is employed to store encrypted data in off chain data repository and asymmetric encryption is used to protect credential attributes during exchange to respective entities with public/private key pair. Key generation service is responsible for generating a single key for symmetric encryption and a public private key pair for asymmetric encryption. Digital signature ensures that the digital credential issued to an entity is legitimate. Using data storage service, the issuer is able to store credentials to off chain data storage (i.e., relational data storage, decentralized file systems, cloud storage etc.) and hash of credential to on chain data storage which confirms integrity of it. Tokenization service allows identity owners to generate consent tokens which can be used only once by verifier.

Application programming interfaces: our framework performs data and service exchange operations using three different APIs. A lightweight and flexible REST API is able to handle multiple types of users' calls over pre-existing protocols like HTTP. Through REST API users can interact with core services to perform basic operations of the

Volume 11, Issue 3
June 2022

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

framework.API libraries such as web3, web3j, Nethereum etc. provide a way to interact with Ethereum smart contracts. Web3, which is written in JavaScript, is used in our framework to facilitate interaction between core services and smart contracts. Interplanetary File System (IPFS), a persistent storage, is capable of storing encrypted credential records in a decentralised manner, overcoming the problem of a single point of failure. IPFS content can be accessed by a unique content identifier which can be generated by a hash function. A set of utility functions of IPFS API enable off chain data service to interact with IPFS.

Smart contracts: Our framework consists of four smart contracts: Enrolment Contract, CredentialIssuance Contract, Consent Contract and Verification Contract. Table 3 displays abbreviations used for smart contracts.

Table 3
Abbreviation for Smart Contract

Abbreviation	**Full Name**
EMC	Enrolment Contract
CIC	Credential Issuance Contract
COC	Consent Contract
VC	Verification Contract

The Enrolment Contract (EMC) is a contract responsible to validate an entity that requests for registration and also maintains a chain for all enrolled entities. EMC classifies enrolled entities with different classes in the system as issuer, identity owner, guardian or verifier. The EMC also maintains Ethereum addresses of entities with public key.

Module 1 shows memory fields and mapping details of EMC smart contract. EMC maintains an array containing the addresses of participants who have been registered for the system. We have defined a data structure named Entity which contains information of a particular entity. Entity comprises Ethereum address, public key to perform cryptographic operations, date and time of registration and class of an entity. Mappings defined in EMC contract provides an efficient way to store and retrieve records based on key value pairs. EMC is able to access complete Entity information using the address of an entity as a key value. Moreover, it provides a list of entity addresses using class as a key value.

Module 1: Entity Enrolment Contract Memory Fields and Mapping Details

Memory Fields:

```
address[] entityChain

structure Entity contains
    address entityAddress
    address proposerAddress
    string publicKey
uinttimeStamp
    string class
```

Mapping:

```
mapping
(address=>Entity)entity
mapping (string=>address[])
entitylisibyclass
```

The EMC sets values of an entity if the sender of this transaction is an administrator node. Last, the enrolment of an entity will trigger the event LogEntityEnrol () to notify the completion status. At this time, the EMC has completed the enrolment, but the credential has not been transferred to the qualified user. Moreover, EMC has getter functions to retrieve required fields of an entity structure and chain of entities.

The CredentialIssuance Contract (CIC) is a contract responsible to maintain details of valid digital credentials of an entity. After verifying attributes, E_{CIS} is able to issue a digital credential (C) to E_{IO} using CIC. Module 2 shows memory fields and mapping details of CIC.

Credential data structure maintains details of a specific digital credential with issuer details.CIC computes unique credentialID for each credential based on timestamp and attribute values. Then with the help of a proposed randomized method, CIC maintains RH to the blockchain. CIC also maintains links to the external data sources. Moreover, E_{CIS} is able to set status as cancelled, suspended or revoked of a specific credential using the status field.

CIC is able to access complete Credential information using credentailID as a key value. Moreover, it provides a list of credentials owned by a particular E_{IO} using entity address as a key value. Last, the issuance of a credential will trigger the event LogCredentialIssuance() to notify the completion status.

Module 2: Credential Issuance Contract Memory Fields and Mapping Details

Memory Fields:

```
structure Credential contains
    string credentialID
    address ownerAddress
    address issuerAddress
    string querylink
    bytes32 rootHash
    string status
```

Volume 11, Issue 3
June 2022

ITEE Journal
Information Technology & Electrical Engineering

ISSN: - 2306-708X

Mapping:	mapping (address=>Entity)entity mapping (string=>address[]) entitylisibyclass

The Consent Contract (COC) is a contract responsible to maintain details of consent for a digital credential of an entity. When E_{IO} wants to share credential attributes to E_{CV}, COC stores a consent token which is required to verify credential attributes with Blockchain.Module 3 shows memory fields and mapping details of COC smart contract.

Module 3: Consent Contract Memory Fields and Mapping Details

Memory Fields:	structure Consent contains string consentID address ownerAddress address verifierAddress string credentialID string token uint validity uint nonce
Mapping:	mapping (string=>Consent) consent mapping (address=>string[]) consentListByOwner

The Consent data structure maintains details of a specific token of a digital credential with verifier details. Tokenization service computes unique Consent Token for each request based on timestamp and verifier address. Using COC, the Identity owner can specify the validity of CT. Moreover, it provides a list of ConsentID shared by a particular entity using entity address as a key value. Last, the consent of a credential will trigger the event LogConsent() to notify the completion status.

The Verification Contract (VC) is a contract responsible to maintain verification status of digital credentials of an entity. When E_{CV} verifies credential attributes with the help of consent token and RH, VC stores a verification status with Blockchain.Module 4 shows memory fields and mapping details of VC smart contract.

Module 4: Verification Contract Memory Fields and Mapping Details

Memory Fields:	structure Verification contains string consentID bool status
Mapping:	mapping (string=>Verification) verification

The Verification data structure maintains details of a verification status of a digital credential. When E_{CV} submits a request to verify a particular attribute or all attributes of a credential, first, VC checks consent token with validity. Last, the verification of a credential will trigger the event LogVerificationStatus() to notify the completion status.

5. EXPERIMENTATION AND RESULTS

To implement our identity framework, in this section we discuss required tools and technologies. (a) Ethereum Geth is an Ethereum client written in Go language which provides a peer to peer network for a decentralized environment. We have deployed smart contracts on Geth to record transactions. (b) Remix is an online tool to compile, deploy and test smart contracts. (c) Solidity is a high level programming language to implement smart contracts. Solidity compiler compiles smart contracts and generates byte codes which run on Ethereum Virtual Machine (EVM) (d) Node.js is a non-blocking and asynchronous programming language which can handle requests from users. Using it we have created dynamic web pages for user interaction. Smart contracts are collections of functions and data like other programming languages. The key difference is smart contracts reside at a specific blockchain address and interaction with them are irreversible. The gas consumption to deploy smart contracts is 917306,1829908,1417391, and 1286807 for EMC, CIC, COC and VC respectively, as depicted in figure 3. Moreover, we analyze the latency, average time to deploy, for the smart contracts, depicted in figure 4.

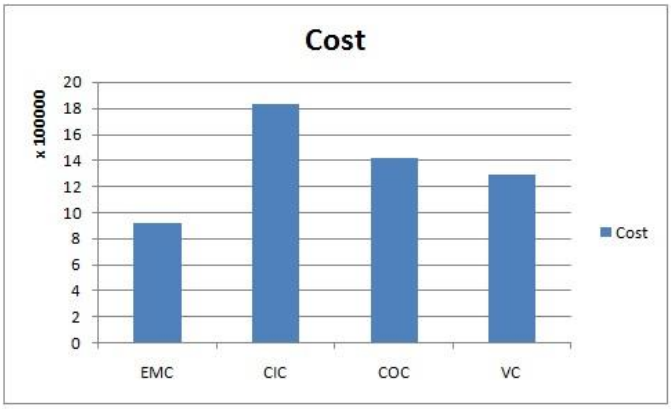

Fig. 3 Smart Contract Deployment Cost

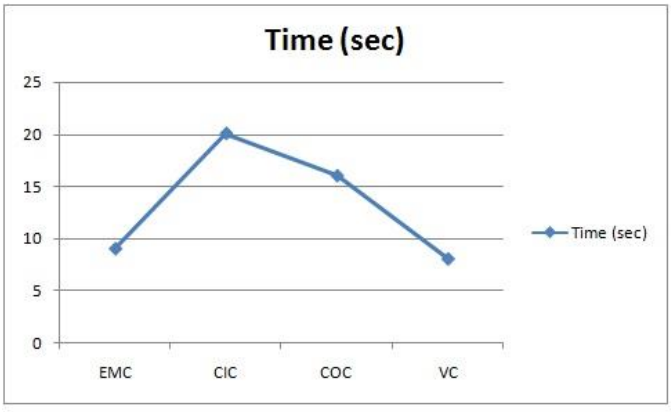

Fig. 4 Latency of Smart Contract Deployment

6. CONCLUSION AND FUTURE WORK

In this research paper, we propose a conceptual framework for identity management with smart contract design and deployment which gives ownership and control of identity attributes to the identity owner to selectively disclose and securely transfer them with a service provider or verifier. The challenges associated with the existing identity systems are security, privacy, ownership, control, and transparency. We focus on achieving transparency and selective disclosure using Blockchain technology using a tree-based verification method. Our conceptual framework provides security and consent of an entity to validate credentials. In the later part of this research, we would like to use formal methods for detailed specification and security analysis. In future, we intend to implement proposed framework modules with a specific use case i.e. education domain for transcript verification.

REFERENCES

[1] Statista. Internet of Things (IoT) connected devices installed base worldwide from 2015 to 2025 (in billions),2019. https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/

[2] A. Tobin and D. Reed, “The inevitable rise of self-sovereign identity,” The Sovrin Foundation, vol. 29, 2016.

[3] Faber, Benedict, et al. "BPDIMS: a blockchain-based personal data and identity management system." (2019).

[4] Prabhakar, Salil, Sharath Pankanti, and Anil K. Jain. "Biometric recognition: Security and privacy concerns." IEEE security & privacy 1.2 (2003): 33-42.

[5] Berghel, Hal. "Equifax and the latest round of identity theft roulette." Computer 50.12 (2017): 72-76.

[6] Isaak, Jim, and Mina J. Hanna. "User data privacy: Facebook, Cambridge Analytica, and privacy protection." Computer 51.8 (2018): 56-59.

[7] Buresh, Donald L. "A Comparison between the European and the American Approaches to Privacy." Indon. J. Int'l & Comp. L. 6 (2019): 257.

[8] Casey, Michael J., and Paul Vigna. "In blockchain we trust." MIT Technology Review 121.3 (2018): 10-16.

[9] Houtan, Bahar, AbdelhakimSenhajiHafid, and DimitriosMakrakis. "A survey on blockchain-based self-sovereign patient identity in healthcare." IEEE Access 8 (2020): 90478-90494.

[10] Stokkink, Quinten, and Johan Pouwelse. "Deployment of a blockchain-based self-sovereign identity." 2018 IEEE international conference on Internet of Things (iThings) and IEEE green computing and communications (GreenCom) and IEEE cyber, physical and social computing (CPSCom) and IEEE smart data (SmartData). IEEE, 2018.

[11] Soltani, Reza, Uyen Trang Nguyen, and Aijun An. "A new approach to client onboarding using self-sovereign identity and distributed ledger." 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData). IEEE, 2018.

[12] Arenas, Rodelio, and Proceso Fernandez. "CredenceLedger: a permissioned blockchain for verifiable academic credentials." 2018 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC). IEEE, 2018.

[13] Mohammed Shuaib, Noor Hafizah Hassan, Sahnius Usman, Shadab Alam, Surbhi Bhatia, Deepika Koundal, Arwa Mashat, Assaye Belay, "Identity Model for Blockchain-Based Land Registry System: A Comparison", Wireless Communications and Mobile Computing, vol. 2022, Article ID 5670714, 17 pages, 2022. https://doi.org/10.1155/2022/5670714.

[14] Reed, D., et al. "Decentralized Identifiers (DIDs) v1. 0 Core Data Model and Syntaxes." W3C First Public Working Draft, URL https://www. w3. org/TR/did-core/[Google Scholar] (2019).

[15] World Wide Web Consortium. "Verifiable credentials data model 1.0: Expressing verifiable information on the web." https://www.w3.org/TR/vc-data-model/?# core-data-model (2019).

[16] Windley, P., and Reed D. Sovrin. "A Protocol and Token for Self-Sovereign Identity and Decentralized Trust." Utah: Sovrin Foundation (2018).

[17] Naik, Nitin, and Paul Jenkins. "uPort open-source identity management system: An assessment of self-sovereign identity and user-centric data platform built on blockchain." 2020 IEEE International Symposium on Systems Engineering (ISSE). IEEE, 2020.

[18] LifeID, “An open-source, blockchain-based platform for self-sovereign identity,” LifeID, Tech. Rep., 2018, [Accessed: 4 - mar - 2022]. [Online]. Available: https://lifeid.io/whitepaper.pdf

[19] SelfKey, The SelfKey Foundation ,[Accessed: 3-feb-2022. [online]Available https://selfkey.org/wp-content/uploads/2019/03/selfkey-whitepaper-en.pdf

[20] Reid, Bob, Brad Witteman, and W. Brad. "Everid whitepaper." EverID, techreport, May (2018).

[21] Takemiya, Makoto, and Bohdan Vanieiev. "Sora identity: Secure, digital identity on the blockchain." 2018 ieee 42nd annual computer software and applications conference (compsac). Vol. 2. IEEE, 2018.

[22] Othman, Asem, and John Callahan. "The Horcrux protocol: a method for decentralized biometric-based self-sovereign identity." 2018 international joint conference on neural networks (IJCNN). IEEE, 2018.

[23] Hamer, Tom, et al. "Private Digital Identity on Blockchain." BlockSW/CKG@ ISWC. 2019.

[24] Al-Bassam, Mustafa. "SCPKI: A smart contract-based PKI and identity system." Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts. 2017. [25] Buresh, Donald L. "A Comparison between the European and the American Approaches to Privacy." Indon. J. Int'l & Comp. L. 6 (2019): 257.

[25] Alsayed Kassem, Jamila, et al. "DNS-IdM: A blockchain identity management system to secure personal data sharing in a network." Applied Sciences 9.15 (2019): 2953.

[26] Sonnino, Alberto, et al. "Coconut: Threshold issuance selective disclosure credentials with to distributed ledgers." arXiv preprint arXiv:1802.07344 (2018).

[27] Halpin, Harry. "Nym credentials: privacy-preserving decentralized identity with blockchains." 2020 Crypto Valley Conference on Blockchain Technology (CVCBT). IEEE, 2020.

[28] Lee, Jeonghyuk, et al. "Sims: Self sovereign identity management system with preserving privacy in blockchain." Cryptology ePrint Archive (2019).

[29] Asamoah, Kwame Omono, et al. "Zero-chain: A blockchain-based identity for digital city operating system." IEEE Internet of Things Journal 7.10 (2020): 10336-10346.

[30] Yang, Xiaohui, and Wenjie Li. "A zero-knowledge-proof-based digital identity management scheme in blockchain." Computers & Security 99 (2020): 102050.

AUTHOR PROFILES

Jayana Kaneriya received M. Tech degree in Computer Science from Sardar Vallabhbhai National Institute of Technology, Surat. She is currently a research scholar at Kadi Sarva Vishwavidyalaya, Gandhinagar. She is working as Assistant Professor in the Department of Computer Engineering at Leelaben Dashrsthbhai Ramdas Patel Institute of Technology, Gandhinagar. She has more than 14 years of academic experience and her research areas include Blockchain Technology, IOT, Network Security, and Distributed Computing.

Dr. Hiren B. Patel is currently working as a Principal of Vidush Somany Institute of Technology and Research, Kadi-Gujarat. He completed his Ph.D. from National Institute of Technology (NIT), Surat with Cloud Computing as the domain of research. He completed his post-graduation with Microprocessor as specialization from M. S. University, Baroda and graduated in Computer Engineering from BVM Engineering College, V.V.Nagar. Having more than 20 years of teaching experience, he has done more than 70 research articles of international repute out of which one was rewarded as best review paper by Elsevier and three more were linked to the United Nations Sustainable Development Goals, helping to tackle some of the world's great challenges. Few of his main subjects of interest are Cloud computing, parallel Processing, Networking and Security and Blockchain Technology.