2018 42nd IEEE International Conference on Computer Software & Applications

Sora Identity: Secure, Digital Identity on the Blockchain

Makoto Takemiya
Soramitsu
Tokyo, Japan
takemiya@soramitsu.co.jp

Bohdan Vanieiev
Soramitsu Labs
Innopolis, Russia
bogdan@soramitsu.co.jp

***Abstract*—Digital identity is the cornerstone of a digital economy. However, proving identity remotely is difficult to do. To complicate things further, identity is usually not a global, absolute construct, but the information shared with different parties differs, based on the relationship to the user. Therefore, a viable solution for digital identity should enable users to have full control over their personal information and share only the information that they wish to share with each service. Blockchain technology can help to realize a self-sovereign identity that puts the user in control of her information, by enabling a decentralized way to handle public key infrastructure. In the current contribution, we present the Sora identity system, which is a mobile app that utilizes blockchain technology to create a secure protocol for storing encrypted personal information, as well as sharing verifiable claims about personal information.**

***Keywords*-Blockchain, distributed consensus, distributed ledger technology, digital identity.**

I. Introduction

Questions of identity permeate human history. In the Torah, Jacob fakes the identity of his brother in order to get a blessing from his father. In more contemporary society, identity theft is rampant, and financial institutions spend significant resources (on average, $60 million per bank [1]) on fulfilling their requirements of knowing their customers (KYC) and related anti-money laundering (AML) requirements. Despite the need for digital identity to support tasks such as asset management and cybersecurity [2], there are still many problems with contemporary solutions to digital identity, as shown by the many hacks that leaked personal information [3], [4].

The present contribution proposes to use blockchain technology, which provides tamper-evident storage of data through using advanced cryptography, as a way to augment contemporary solutions for identity management. We propose that blockchain technology can increase the security of identity management systems by decentralizing the structure of the system so that a single point of trust is not needed, thus increasing resilience against attacks. Cryptography can be used to securely give users direct control over their own information, with low probability of being hacked and leaked.

II. Related work

Contemporary solutions for digital identity are centered around public key infrastructure (PKI) [5]. However, there are problems of trust associated with traditional PKI that relies on centralized certificate authorities [6]. Additionally, many centralized services that manage a user's information suffer from systemic security risks that often leak a user's personal information [7].

An alternative to centrally managed identity is a self-sovereign identity, where an individual has full control over their information, security assurances provide that the privacy of the information is not compromised, and the individual also controls who has access to their personal information [8].

Often blockchain technology is thought of as being associated with cryptocurrencies, such as Bitcoin [9]. However, the same technology that allows any participant in a blockchain network to verify data in the blockchain also allows the verification of other data, such as verifiable claims about identity. In the system proposed by Baars [10], because anyone can verify that authenticity of digital signatures in a blockchain system, a decentralized system for sharing claims about the identity of individuals can be created.

Providing claims about users in a blockchain system functions as a notary, only instead of having designated authorities providing notarization, anyone can sign data and anyone can choose which data signatories they wish to trust. For example, Estonia is using blockchain technology for notarizing personal information associated with their E-residency program [11], so anyone who wishes to trust the information from the E-residency program can verify it by comparing the signatures of the information in the blockchain. Moyano and Ross [1] similarly propose to use blockchain technology to streamline KYC procedures for banks by sharing verifiable proof of validation of a user's information.

To work towards standards to enable a self-sovereign digital identity, the W3C has proposed Decentralized Identifiers (DIDs)[1]. DIDs are globally unique identifiers that use cryptography to generate proofs of data that can be resolved

[1] https://w3c-ccg.github.io/did-spec

0730-3157/18/$31.00 ©2018 IEEE
DOI 10.1109/COMPSAC.2018.10299

using the identifiers. The DIDs can provide a standardized format for sharing of verifiable claims about a user's identity.

In the present contribution, we propose to use key-value stores that are encrypted with a cryptographic key that is owned by the user, and hashes of the values of personal information are salted and put into a blockchain platform. The Sora mobile apps allow the user to generate their cryptographic key, input their data, encrypt it, and publish salted hashes of their data to the blockchain. Users can then share their personal information of their own volition to institutions, such as banks or other corporations, and those institutions can, in turn, cryptographically sign hashes of salted personal information, thus acting as a notary.

III. Motivating Example

Consider a user, U, who uses the Sora identity mobile app (Section V) to manage their identity. They install the app and enter their personal information. Their personal information is then salted and digitally signed by them (Section IV) and the digital signature to prove that they approve the information is entered into a blockchain. Then, they share their personal information with an institution, I, who then digitally signs a salted hash of the information and publishes the hash and signature to a blockchain. Another user, U, who wants to open an account with institution, I_2, then receives personal information from U and then verifies the signature from I to confirm that U's information was verified.

In this example, the Sora identity mobile app works to realize a self-sovereign digital identity solution by giving the user, U, control over their personal data, as the user can encrypt their data and send it to an institution themselves. Once information is shared, institutions can then notarize the personal information by putting signed hashes on the blockchain. These digitally signed hashes can then form the basis for verifiable claims[2] of identity.

Another example includes an issuer I, a user U, a verifier V. I is able to create verifiable claims of any kind (issue document, health record, and many more) about U without disclosing their contents by uploading signed salted hashes to a blockchain. U can then share claims with V, which can cryptographically verify their content.

IV. Sora Identity Protocol

This section describes the Sora Identity Protocol, the core component of this contribution. Our approach here is *top-down*, *id est*, we give a top level view on the system and then provide more details on each component.

The main actors in the protocol are the User, her Mobile Device, a central Server, and a blockchain platform (e.g. Hyperledger Iroha).

Digital identity can be represented as a set of key-value bindings–*entries*. An example of two entries are `key=birthday, value=01/01/1962; key=first-name, value=Bohdan`. To express semantics of the "keys" of digital identity, we use JSON-LD[3].

Every user of the system has a unique identifier—DID (Decentralized Identifier) [12]. DIDs and the corresponding Document Objects (DDOs; example is shown in listing 2) are managed through a decentralized DID Resolver, that is capable of doing CRUD operations on DDOs. This resolver implements the Sora Method Specification[4] and is based on Hyperledger Iroha. Users specify their public keys in the DDO and use the corresponding keypairs to create transactions for Iroha.

To prevent the loss of keypairs and allow easy migration, every keypair is stored on a central Server in encrypted form, such that only the keypair Owner can decrypt them.

It should be noted that users' private data can be stored on a remote Server in an encrypted form, without the Server having the ability to decrypt it (Section IV-B).

Any user with a DID is able to issue a verifiable claim[5] about themselves or other users. Often such claims will contain private information, such as the details of digital identity. To prevent data loss we separate Verifiable Claims into two parts. The first part is public (listing 4)—it is stored on the blockchain and consists of salted hashes of the claims themselves, a digital signature, and information about issuer. The second part is private (listing 3)—it is shared with the Verifier, when requested.

The public part of the Verifiable Claim in combination with the blockchain is used to achieve:

- privacy—preimage of hashed digital identity (claim) can not be disclosed without the Owner's permission
- non-repudiation—a User can not silently change digital identity or repudiate its ownership
- selectivity—a User is free to choose, what part of identity should be shared
- integrity—a hashed identity can not be altered or changed
- time-locking—using a blockchain, it is easy to prove that an identity was created or modified at a certain time, or revoked (and therefore, invalid) after certain time
- pseudonymity—a User can create an arbitrary number of DIDs to avoid correlating different parts of their digital identity.

For example, a user may issue 5 claims, one for every passport field, and combine them into a single verifiable claim. If a Verifier needs only the birth date, the Prover (claim subject) can provide only one, corresponding claim, and the Verifier can simply calculate the hash and make the

[2]https://www.w3.org/TR/verifiable-claims-use-cases/

[3]http://json-ld.org/

[4]https://github.com/soramitsu/did-spec

[5]https://github.com/soramitsu/vc-data-model

hash lookup on a public part of claim. The Verifier can also add a signature to the public part of the claim thus attesting to its validity.

A. User-Side Secrets and Authentication

To ensure that only a user is able to decrypt her identity, a user selects a Personal Password, (P), that has the lowest bound for security: 48 bits of entropy; this corresponds to an 8-digit password which has capital letter(s), small letter(s), and digits.

A user must generate a kseypair, which is used to interact with the blockchain. Since we use Hyperledger Iroha as the blockchain platform for storage of notarized personal information, a user needs to generate an Ed25519/SHA3-512 [13] keypair. This keypair is encrypted with a user's data encryption key, K (Section IV-C), which in turn, is derived from P, so that even if an attacker steals this keypair, she cannot use it without P.

Thus, P becomes a master password, allowing the user to decrypt her digital identity and all necessary meta information, including the keypair.

B. Entry Creation and Storage

By *entry*, we assume a single key-value binding, such as `key=birthday, value=01/01/1962`.

We describe the algorithm by example. Lets assume a user wants to add her birth date, then she follows the algorithm:

1) generate $salt$—a 16-byte random string
2) a claim is created (listing 3), it may look like the following JSON:

Listing 1: Claim Example

```
{
  "id": "<did>",
  "birthday": "01/01/1962",
  "salt": "<16 bytes>"
}
```

   Then, this JSON-LD is serialized using the JSON-LD normalization algorithm, `URDNA2015`, and hashed with `SHA3-256`. This guarantees that nobody is able to calculate a preimage of the hash.
3) encrypt claim with a block cipher, such as `AES256`, using the key K (Section IV-C) and store it locally on the mobile phone. Thus, the User is the only Owner of these data.
4) create a verifiable claim with all the necessary meta information, including the hash of the claim
5) use any keypair to create a transaction and broadcast it to the Iroha blockchain. This transaction contains the public part of the verifiable claim.

C. Data Encryption Key Derivation

A user's password, P, can potentially be weak and easy to guess with a dictionary attack. We cannot protect it from a brute force attack, but we can make it hard enough to be unrealistic to brute force. For this we use a Password Based Key Derivation Function (PBKDF2 [14]). PBKDF is defined as:

$$K = PBKDF2(PRF, Password, Salt, c, dkLen)$$

where:

- `PRF` is SHA3-512
- `Password` is the User's Password, P
- `Salt` is

  $$Salt = SHA3-512(\text{“}SORA-SALT'' + P)$$

  (SHA3-512 of string “SORA-SALT” concatenated to the User's Password, P)
- `c` is 61337 (number of rounds)
- `dkLen` is 256 (output Key is 256 bits long)

The resulting output key is called the Data Encryption Key, K, and is 256 bits long.

D. Identity Sharing and Notarization

A user can notarize another user's identity by adding her signature to the blockchain.

We demonstrate the notarization protocol via example. Consider two users: A and B.

1) A publishes a verifiable claim on the blockchain.
2) B requests a certain attribute of digital identity.
3) A shares a preimage of the hash with the claim.
4) B calculates the hash and compares it with one of the hashes from the verifiable claim, found on a blockchain.
5) If a hash matches, the signature is valid, and the verifiable claim has not been revoked, identity is verified.

User B can then “notarize” this claim by creating a signature for this Verifiable Claim. The signature can be sent back to the user A, who publishes it to her account.

V. Sora Identity Mobile Application

To realize our goal of a secure, digital identity, we implemented a mobile application for Android and iOS, that interacts with a permissioned, Hyperledger Iroha[6] blockchain to store proofs of verifiable claims about users' personal information, in accordance with the Sora Identity Protocol.

Figure 1 shows a screen from the Sora Identity application. Using the mobile application, users can enter their personal information and upload salted hashes to the Iroha blockchain.

The Sora Identity mobile application is a proof-of-concept demonstrating that the Sora Identity Protocol can be realized

[6]https://github.com/hyperledger/iroha

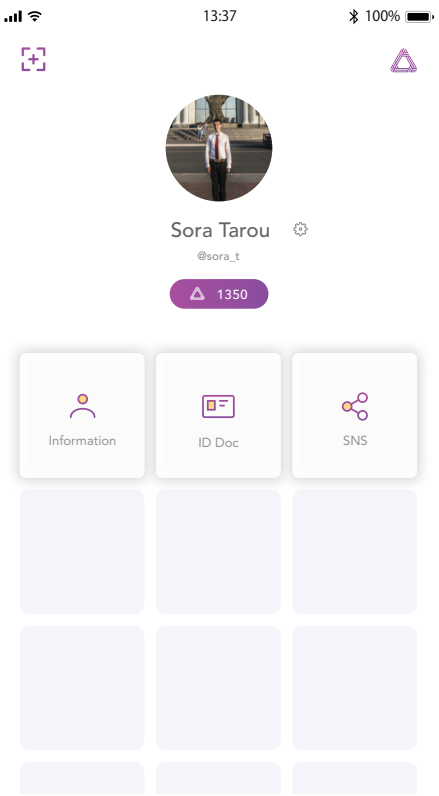

Figure 1: *Sora Identity Mobile App* Home screen where user can choose to edit or share their personal information from.

in a mobile environment, demonstrating a proof-of-existence for the viability of our protocol.

VI. Conclusion

We presented the Sora identity system to manage decentralized, self-sovereign identities on the blockchain. The Sora Identity Protocol presented in this contribution outlines a practical, production-ready method for managing decentralized, verifiable claims about identity using a blockchain system.

Our system uses mobile applications to allow users to interact with the permissioned blockchain, Hyperledger Iroha, in order to digitally sign and share proofs of their personal information. The mobile applications demonstrate that our protocol is feasible to implement in production.

For future work, we want to implement more tools to allow institutions to verify a user's information and share their notarized information with others.

Acknowledgment

Special thanks to Bulat Mukhutdinov for his comments on the protocol presented in this paper.

References

[1] J. P. Moyano and O. Ross, "Kyc optimization using distributed ledger technology," *Business & Information Systems Engineering*, vol. 59, no. 6, pp. 411–423, 2017.

[2] A. R. Friedman and L. D. Wagoner, "The need for digital identity in cyberspace operations," *Journal of Information Warfare*, vol. 14, no. 2, pp. 42–52, 2015.

[3] M. Levine and J. Date, "22 million affected by opm hack, officials say," *ABC News, July*, vol. 9, 2015.

[4] J. Silver-Greenberg, M. Goldstein, and N. Perlroth, "Jpmorgan chase hack affects 76 million households," *New York Times*, vol. 2, 2014.

[5] U. Maurer, "Modelling a public-key infrastructure," in *European Symposium on Research in Computer Security*. Springer, 1996, pp. 325–350.

[6] C. Ellison and B. Schneier, "Ten risks of pki: What you're not being told about public key infrastructure," *Comput Secur J*, vol. 16, no. 1, pp. 1–7, 2000.

[7] D. Mulligan and A. Schwartz, "Your place or mine?: privacy concerns and solutions for server and client-side storage of personal information," in *Proceedings of the tenth conference on Computers, freedom and privacy: challenging the assumptions*. ACM, 2000, pp. 81–84.

[8] A. Tobin and D. Reed, "The inevitable rise of self-sovereign identity," *The Sovrin Foundation*, 2016.

[9] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[10] D. Baars, "Towards self-sovereign identity using blockchain technology," Master's thesis, University of Twente, 2016.

[11] C. Sullivan and E. Burger, "E-residency and blockchain," *Computer Law & Security Review*, vol. 33, no. 4, pp. 470–481, 2017.

[12] W3C, "DID Specification," https://w3c-ccg.github.io/did-spec/, 2018.

[13] Hyperledger, "Iroha-Ed25519," https://github.com/hyperledger/iroha-ed25519, 2018, [Online; accessed 16-April-2018].

[14] Wikipedia, "Password Based Key Derivation Function 2," https://en.wikipedia.org/wiki/PBKDF2, 2018, [Online; accessed 16-April-2018].

Listing 2: DID Example

```
{
  "@context": [
    "https://example.com/vocabulary",
    "http://example.com/contexts/security"
  ],
  "id": "did:sora:iroha:bogdan@soramitsu.co.jp",

  "publicKey": [{
    "id": "did:sora:iroha:bogdan@soramitsu.co.jp#keys-1",
    "type": "RsaVerificationKey2018",
    "owner": "did:sora:iroha:bogdan@soramitsu.co.jp",
    "publicKeyPem": "-----BEGIN PUBLIC KEY...END PUBLIC KEY-----\r\n"
  }, {
    "id": "did:sora:iroha:bogdan@soramitsu.co.jp#keys-2",
    "type": "RsaEncryptionKey2018",
    "owner": "did:sora:iroha:bogdan@soramitsu.co.jp",
    "publicKeyPem": "-----BEGIN PUBLIC KEY...END PUBLIC KEY-----\r\n"
  }, {
    "id": "did:sora:iroha:bogdan@soramitsu.co.jp#keys-3",
    "type": "Ed25519VerificationKey2018",
    "owner": "did:sora:iroha:bogdan@soramitsu.co.jp",
    "publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAjZpfkcJCwDwnZn6z3wXmqPV"
  }],

  "authentication": [{
    "type": "RsaSignatureAuthentication2018",
    "publicKey": "did:sora:iroha:bogdan@soramitsu.co.jp#keys-1"
  }],

  "created": "2002-10-10T17:00:00Z",

  "proof": {
    "type": "Ed25519Signature2018",
    "created": "2002-10-10T17:00:00Z",
    "creator": "did:sora:iroha:bogdan@soramitsu.co.jp#keys-3",
    "signatureValue": "QNB13Y7Q9...1tzjn4w=="
  }
}
```

Listing 3: Verifiable Claim - private part

```
{
  "claim": [{
      "id": "did:sora:iroha:bogdan@soramitsu.co.jp",
      "passportNumber": "FB216121",
      "salt": "salt1"
    }, {
      "id": "did:sora:iroha:bogdan@soramitsu.co.jp",
      "name": "Bogdan",
      "salt": "salt2"
  }]
}
```

Listing 4: Verifiable Claim - public part

```
{
  "@context": "https://example.com/contexts/Credential",
  "id": "fe8f1f4d-d971-4e2b-8633-4a31c856fe04",
  "issuer": "did:sora:iroha:bogdan@soramitsu.co.jp",
  "issued": "2010-01-01T19:73:24Z",
  "claim": [
    "ffb9f9f3f6...8bcff2abbc21c7",
    "9b1d60ad74...98573d124d04b1",
    "deeee9e2e4...f8742bae293d0c"
  ],
  "proof": {
    "type": "LinkedDataSignature2015",
    "created": "2018-02-08T16:02:20Z",
    "creator": "did:sora:iroha:bogdan@soramitsu.co.jp#keys-1",
    "signatureValue": "QNB13Y7Q9...1tzjn4w=="
  }
}
```