Industry-Grade Self-Sovereign Identity

MSc Thesis

Rowdy Chotkan
R.M.Chotkan@student.tudelft.nl

Introduction

This document describes research into the development of an *Industry-Grade Self-Sovereign Identity* (IG-SSI) scheme. This scheme will be developed with collaboration of the Dutch Ministry of the Interior and Kingdom Relations and will serve as research into a digital identity scheme for the European Union. As this thesis is written per requirements of the 4TU Cyber Security programme, it will focus on applicable Cyber Security concepts and as such privacy and security will be the core of the design.

Problem Statement

There exist two reasonings for the rationale of Self-Sovereign Identity's existence: the first reasoning is to devoid the current oligopoly of big-tech companies in the digital identity domain. The main issues regarding this oligopoly are lack of control, privacy. and information asymmetries. The foremost issue is lack of control: the service provider may revoke ones digital identity without warning, resulting in a loss of access to possibly countless of services. SSI attempts to resolve this issue by allowing the digital identity to be owned by the subject theirself.

These identity providers are essentially commercial parties, profiting from data received through managing these identities. This breach of privacy often comes hand in hand with the free to use service offered by the digital identity service providers. The often circulating quote "If you are not paying for it, you're not the customer; you're the product being sold"[1] holds up in this regard. The issue with commercially available identities is that they do not provide legally valid identities and pose a huge threat on privacy, as the subject has no control over with whom their data is shared. This additionally leads to information asymmetries: as these big-tech companies posses large amount of PII of their users, any economic transaction made with them, results in them possessing more knowledge than the buyer. This effect has been regarded by Tobin and Reed (2016) as the use of adhesion contracts, which go against the users' best interests.

The second reasoning is economic inclusion: residents residing in countries devoid of proper (central) identity infrastructure, are excluded from essential services enabled through identification system. **?** defines identification to be required for the following:

- Inclusion and access to essential services: e.g., healthcare, education, and financial services.
- Effective and efficient administration of public services, policy decisions and governance.
- Accurate measure of development progress in areas.

Hence, without any form of valid identification measures, these residents are devoid of essential services and are less likely to be able to improve their living conditions or receive aid.

As the first issue, mostly regarding privacy and control, is a far more relevant topic in Computer Science, with the second problem is more a socio-politic issue, the primary focus of this research will be targeted at combating the former phenomena.

Background Information

History of Identity Management Systems

As described by Allen (2016), there exist four different phases in identity management systems. Next, we describe each of them.

Phase One: Centralised Identity With the onset of the Internet, centralised authorities such as IANA and ICANN became the issuers and authenticators of digital identities. E.g., the IANA determined the validity of IP addresses. Next, in order to generate trust through certificates, Certificate Authorities were created, which were able to also delegate some power through hierarchies. Finally, as mentioned by Cameron (2005), the distributed nature of the internet let to each platform implementing its own digital identity management in the form of e.g. user accounts. All of the above properties of the current Internet ecosystem are inherently centralised authorities. With the consequence of the user not

[1] https://www.metafilter.com/95152/Userdriven-discontent#32560467

owning any of his digital identities, as their are all either assigned to her or managed by others. Already in 1991, Zimmermann (1999) showed that distributed identity management is indeed possible, to some extend. However, Zimmermann's PGP was never widely adopted.

Phase Two: Federated Identity

The second generation of attempted to overcome the hierarchies, by imagining a *federated identity*. An example of this is Microsoft's Passport initiative, allowing identities across different domains, in this case, multiple websites. However, this initiative soon proved to be far from optimal, as it makes Microsoft the main authority. This was improved upon by allowing each site to remain an authority.

Phase Three: User-Centric Identity

The third generation attempts to put the user at the center of the identity. Examples of these include OpenID[2], OAuth[3] and FIDO[4]. The main goal of these implementation can be said to be user consent and interoperability, as the user has to provide consent for signing in on another domain using the methodology and they can be supported by any domain. However, the main drawback to these solutions are that the registering authorities can withdraw the digital identity at any time and, as such, there is still much to desire for user control.

Phase Four: Self-Sovereign Identity

The above limitations and designs failed to put the control in the user's hands. SSI aims to bridge this gap, by fully decentralising digital identities to such an extent that the user is in full control on what data is stored, what happens with said data, and with whom said data is shared.

Self-Sovereign Identity Infrastructures

Broadly speaking, there exist two kinds of blockchain-based SSI infrastructures: (1) the Identifier Registry Model (IRM) and the Claim Registry Model (CRM) (Mühle, Grüner, Gayvoronskaya, & Meinel, 2018). The

Properties

As no consensus on a formal definition of Self-Sovereign Identity has been reached, the properties of SSI are loosely defined. There are, however, there are returning concepts in (academic) literature and common notions of use-cases. This section will aid in defining a set of requirements based on identified common themes in literature and will bridge the gap in unresolved issues.

One of the foremost motivation behind SSI, is its ability to generate trust in cyberspace. As presented by Cameron (2005), the Internet was built without an identity layer: there is no standardisation for authentication, authorisation and identification. As a consequence, the Internet consists of numerous workarounds of identification, which, evidently, has grown into a oligopoly of identity management held by large organisation such as Google, Apple, and Microsoft. The drawbacks of the current construction are quite broad:

Firstly, the data behind the identification measures, are not in the hands of the users. As a consequence, a user must ask permission to alter his data, has no direct access to his data, and has no control over how his data is processed. As these identities are managed by commercial parties they are often prone to being processed and mined for the gain of said parties. Secondly, as these large organisation are no governmental entities, the resulting identities can never be used for legal identification purposes, an inherent shortcoming of their design. Finally, apart from the trivial overhead in different identification "workarounds", the lack of open-standards and centralised storage often leave such credentials in peril. As often no proper security requirements are set in place (e.g., a simple password), the credentials can either be easily brute-forced or stolen, resulting in identity loss. Their often centralised nature, can be weakness as well, as a security breach may impact the digital identities of all users.

The foremost common theme which can be said to have reached consensus, is the user-centric approach of SSI. Namely, the rationale of SSI's existence is making the user the manager of his own identity.

The most commonly discussed set of properties is that posed by Allen (2016). Allen posed the following set of *principles*, which are to ensure the user-centric nature of SSI. These consist of the following

1. **Existence**: users must have an independent existence. I.e., a (digital) sovereign identity does not solely exist digitally. As a result, it can be interpreted as requiring to be tied to a physical entity.

2. **Control**: users must have control over their identities. This entails a full authority over the user's own identity: the ability to share, update, and even hide.

3. **Access**: users must have access to their own data. Similarly to the above principle, users must be able to access all of their data.

4. **Transparency**: all involving systems and algorithms must be transparent. This entails open-standards and open-source software.

5. **Persistence**: identities must be long-lived. Identities should, thus, exists until destroyed by the user.

[2]For *OpenID*, see `https://openid.net/connect/`

[3]For *OAuth*, see `https://oauth.net/`

[4]For *FIDO*, see `https://fidoalliance.org/`

6. **Portability**: information and services about identity must be transportable. I.e., identities must not be held by a single third-party, as they may not support it livelong. This principle would be satisfied by the *Control* and *Persistence* principles.

7. **Interoperability**: identities must be as widely usable as possible. This ensures that the identities can be globally deployed and can be achieved partly by adopting the *Transparency* principle.

8. **Consent**: users must agree to the use of their identity. This principle strengthens the *Control* principle, as sharing of attributes may only occur with the consent of the user. However, the Allen noted that this must not require interactivity.

9. **Minimalisation**: disclose of claims must be minimised. I.e., the minimal amount of information must be disclosed when sharing claims. This principle is focused on privacy and prevents misuse of data.

10. **Protection**: the rights of users must be protected. The right of users must take precedence over the identity network itself. This can be achieved thorough the *Transparency* principle and decentralisation.

The above set of principles is often adhered to as a set of requirements. See e.g. . These principles portray that digital identities must be tied In addition to these ten principles, Stokkink and Pouwelse (2018) add the principle of *Provability*: claims must be provable, as otherwise they can be deemed worthless.Tobin and Reed (2016) build upon these ten principles by subdividing these into three categories:

- **Security**: aims to keep the digital identity information secure. This consists of: *Protection*, *Persistence*, and *Minimisation*
- *Controllability*: focuses on the user-centric foundation of SSI. This consists of: *Existence*, *Persistence*, *Control*, and *Consent.*
- *Portability*: this requirement results in the user not being tied to a single provider and being able to use their identity without bounds. This consist of: *Interoperability*, *Transparency*, and *Access*.

The additional principle defined by Stokkink and Pouwelse (2018) can be categorised into *Security*, as the provability of claims aids in generating trust and in authentication.

The work set out by Cameron (2005), is another commonly cited set of principles for SSI. In their work, Cameron developed the so-called *Laws of Identity*. These laws explain the shortcomings and successes of digital identity systems and, as such, are applicable to SSI. These consist of the following:

1. **User control and consent**: digital identity systems must only reveal personal identifiable information (PII) given prior consent by the user. Through this law, trust can be built between the system and the user.

2. **Minimal disclosure for a constrained use**: the solution which discloses the least amount of and best limits the use of PII, is the most stable long term solution. This law minimises risk, as it is assumed that a breach is always possible.

3. **Justifiable parties**: disclosure of data with third parties must always be justifiable in a given identity relationship. Through this law, the user is aware of any third parties with whom is interacted with whilst sharing information.

4. **Directed Identity**: universal digital identity systems must support “omni-directional" identifier, which can be said to be public, and “unidirectional" identifiers, which can be said to be private, enabling identification whilst facilitating privacy.

5. **Pluralism of operators and technologies**: universal identity system must support multiple identity technologies run by multiple identity providers. This law enables the incorporate this somewhere, disallowing vendor lock-in and encourages the use of open-standards.

6. **Human integration**: universal digital identity systems must incorporate the user as a component of the system, offering protection against identity attacks. This laws attempts to bridge the discontinuity between the actual (human) users and machines with which they communicate.

7. **Consistent experience across context**: universal digital identity systems must allow for a separations of domains, whilst enabling a consistent experiences within and across them. This law thus enables interoperability across different operators and technologies.

This chapter describes the design of the *Industry-Grade Self-Sovereign Identity Framework* (IG-SSIF).

In its essence, the main enables of self-sovereign identity are *attestations*: the verifiable claims capable of facilitating one’s digital identity. All proposed solutions focus on public-key encryption. The selection of asymmetric encryption as opposed to symmetric encryption lays in the properties that stem from asymmetric encryption. The public and private

key pair enable the possibility to encrypt a message for a certain public key, for which it is then certain that only the entity possessing the corresponding private key, has the ability to decrypt said message. Vice verse, encrypting a message with a private key, ensures that only the corresponding public key can be used for decryption. This first property enables privacy, as only the entity to which the public key belongs, can now read the contents of the message. The second property enables authenticity, as anyone can verify, using the corresponding public key, that a message was signed by a certain entity to which the private key belonging to the public key is known. This verifiability through public-keys allows for a relatively trivial implementation of attestations.

More specifically, the properties of public key encryption can prove to create a rather trivial creation of attestations: one can simply hash a specific piece of information and encrypt it using his private key. This process is referred to as creating a digital signature. Next, anyone possessing the corresponding public key can verify this signature in case he knows the corresponding plaintext value.

Existing Solutions

Sovrin

The Sovrin Foundation[5], focused on creating an identity layer for the Internet, notes several effects caused by the lack of identity management on the Internet. The traditional methodology for identification, i.e. unique credentials for each digital service, creates several layers of problematic side effects. Sovrin note that it is both problematic from a usability perspective and from a security perspective(Tobin & Reed, 2016). Firstly, from a usability perspective, managing different credentials for each service becomes problematic as users often do not take proper security measures. Secondly, the numerous storage location for these fragmented digital identities can prove to be honeypots for hackers, after which a possible breach affects the trust in said service and possible affects the security of a users' other credentials due to the aforementioned lack of proper security measures set into effect by the user theirself. The second phase in identity management, the so-called federated model mentioned byAllen (2016) is also sub-optimal. It foremost increased data leakage through sharing, raising privacy concerns, whilst still not allowing identity management by the user (Tobin & Reed, 2016).

Furthermore, the impact of a missing identity layer causes large financial impacts. Services have to construct their own identity management system and they suffer from fake users, whilst user suffer from stolen records and identity theft (Tobin & Reed, 2016).

Sovrin proposes the use of public permissioned blockchain, consisting of "Members" and "Stewards". Where the former are the user registered with their digital identity and the latter the verifying nodes. The foundation itself is to be tasked with developing, coordinating, governing and promoting the identity network (Tobin & Reed, 2016). They propose the use of two layers of nodes, where the nodes in the outer layer are deemed "Observer Nodes" which run read-only copies of the blockchain, and the inner layer consists of "Validator nodes" which allow for write access (*Sovrin ™ : A Protocol and Token for Self-Sovereign Identity and Decentralized Trust A White Paper from the Sovrin Foundation*, 2018). The reasoning behind this design choice is scalability. The principle is the same as the general concept of SSI: claims are cryptographically signed for a user, after which these can be verified by third-parties. Sovrin aims to store no private (encrypted) data on its blockchain. Additionally, Sovrin is compatible with the DID[6] standard from W3C (Reed, Law, & Hardman, 2016). In order to aid privacy, each relation uses new public and private keys

find citation

Serto (uPort)

. Serto[7], formally known as uPort, is an SSI solution built on Ethereum (Lundkvist, Heck, Torstensson, Mitton, & Sena, n.d.; **?**). Serto has a multitude of open standing project, of which their Ethereum SSI project appears to have gained the most traction. As was the case for Sovrin, Serto is being built to be compatible with the DID standard from W3C. Sovrin is built upon the concept of Ethereum smart contracts, where an identity can be represented by a smart contract of Ethereum address. The usage of Ethereum contracts, make Serto a Claim Registry Models. The contracts store the hashes of claims, of which the claims themselves are stored off-chain (**?**). The underlying structures are built on the JSON format.

Decentralized Identifiers

The aforementioned solutions all utilise W3C's Decentralized Identifiers (DIDs). DIDs are a type of identifier that allow for verifiable, decentralised digital identities (*Decentralized Identifiers (DIDs) v1.0*, n.d.). It is a specification drafted by the World Wide Web Consortium (W3C)[8]. And, being a specification, DIDs have no specific software or hardware requirements, it merely defines a generic syntax and generic requirements for the four CRUD (create, read, update, delete) operations *Decentralized Identifiers (DIDs) v1.0* (n.d.). The design goals of DID are the following *Decentralized Identifiers (DIDs) v1.0* (n.d.):

[5]For *Sovrin*, see `https://sovrin.org/`

[6]For *DID*, see `https://www.w3.org/TR/did-core/`

[7]For *Serto*, see `https://www.serto.id/`

[8]For *World Wide Web Consortium*, see `http://w3.org/`

- Decentralization
- Control
- Privacy
- Security
- Proof-based
- Discoverability
- Interoperability
- Portability
- Simplicity
- Extensibility

The basic structure of DIDs consist of a DID which references a DID documents. The DID documents contains the actual information regarding identification.

Related Works

Mühle et al. (2018)

Mühle et al. (2018) describe an overview of SSI. They state that ISS differentiates itself with traditional identity management systems by being a user centric model as opposed to service provider centric. They describe two architectures for SSI: the *Identifier Registry Model* and the *Claim Registry Model*. Wherein the former model the pairing of identifiers and public keys of users are stored onchain and claims offchain. In the later model, in addition to serving as a registry for identifiers and public keys, the claims themselves are also stored onchain. Next, they focus what they deem the four core components of SSI: identification, authentication, verifiable claims, and attribute storage. Identification comes done to the issue of having both uniqueness and human-readability in identifiers of clients. It is noted that the current best effort is that of *decentralised identified* (DID), which has a universal resolver by the Decentralized Identity Foundation[9]. They present a scheme capable of incorporating the four core components. The resulting scheme satisfies the ten principles byAllen (2016) and presents SSI in a intuitive fashion. The scheme sets verifiable claims at the centre: the these claims are issued by an issuer on a subject, which can be attested by other clients. These signed claims can then be verified by a verifier to whom a claim is presented to.

Der, Jähnichen, and Sürmeli (2017)

Der et al. (2017) describe the o opportunities and challenges for a digital revolution caused by SSI. The authors start with explaining the terms *digital identities* and *secure digital identities*. Where a *digital identity* is a temporal reflection of a regular identity: it merely contains specific characteristics of an identity, with varying level of detail. A digital identity can be held by any type of entity, may it be a person, a car, or a device. It usually has to function to use a particular service. In addition, a *secure digital identity* adheres to the requirements of *privacy* and *trustworthiness*. Where privacy leads to only authorised access to the identity, and trustworthiness the correctness of the attributes contained in the digital identity.

The authors then explain the general concept of Self-Sovereign Identity. They state that SSI can be the next step in identity management and mention the ten principles by Allen (2016). SSI moves the requirements of privacy and trustworthiness to the user, requiring the user to provide evidence.

Next, three opportunities for SSI are explained. Firstly, SSI can counteract the oligopoly present in the management of current digital identities. Secondly, it can provide help to people living in crisis areas, as identities may no longer require ties to local government. Finally, SSI may help companies to adhere to the GDPR as privacy can be more easily implemented.

The challenges for SSI are also explained. It is stated that current digital identity services (e.g. Facebook connect) allow for a certain level of comfort by trading in a certain level of control of their identity. Based on that assumption, the case is made that one of the core challenges of SSI is that the additional required administrative efforts of SSI must be sufficiently comfortable. The following key challenges are outlined:

- Protection of privacy across transactions.
- Transparency between two parties during a transaction, i.e., consensus on content and conduct.
- Persistency of digital identities and logs for long-term transparency.
- Trustworthiness of digital identities and claims.
- Consistency between granted rights and real usage.
- Standardisation of data formations and interfaces.

Finally, the efforts by the ISÆN and an outlook are given with applications of SSI for the Internet of Things and institutions.

Stokkink and Pouwelse (2018)

Stokkink and Pouwelse (2018) present a blockchain-based digital identity solution. It is stated to be an academically pure model for SSI. They state that the first half of the problem regarding the creation of such a model, is the need for

[9] `https://identity.foundation/`

Self-Sovereign Identity: identity holders must be identity owners. The second half of the problem is the need for legally valid signatures: identities can e.g. be recognised by the governments, making them legally valid. They firstly describe the solution for the first halve of the problem, in which they state the ten principles by Allen (2016). The blockchain-nature of their solution is said to intrinsically satisfy the majority of the principles, apart from:

- Portability
- Interoperability
- Minimalisation
- Protection
- Provability (added by authors)

The usage of zero-knowledge proofs and the chain of claims enabled by their blockchain, Trustchain, allows for the satisfaction of the remaining principles. Their solution comprises of zero-knowledge proofs also allowing for range proofs. Their claim metadata incorporates a validity term for finite claim validity as well as a “proof format” field, allowing for interchangeable signature algorithms. A reference implementation shows sub-second claim-verification performance.

Othman and Callahan (2018)

Othman and Callahan (2018) describe their Horcrux protocol, a decentralised biometric credential storage option via blockchain using W3C’s Decentralised Identifiers (DID). The authors mention that the current drawback of traditional biometric-based authentication systems is that the systems are a single point of compromise for securing digital identities. This is caused by requiring a central authority for storing templates of biometric samples. The Horcrux protocol combines the SSI ecosystem with the h 2410-2017 IEEE Biometric Open Protocol Standard (BOPS). This is performed by dividing biometric templates into $n \leq 2$ shares, which are then stored distributed-wise. The actual shares are stored offchain, but resolvers to the DIDs are stored on onchain. Their solutions requires interaction with these BOPS-servers for enrolment into the SSI system.

Ferdous, Chowdhury, and Alassafi (2019)

Ferdous et al. (2019) describe a mathematical model for SSI in order to provide a formal and rigorous treatment of the concept of SSI itself. As such, they firstly formalise a mathematical definition and identify the required properties for SSI, after which they investigate the impact SSI can have using the Laws of Identity. Finally, they investigate the implication of applying blockchain technology to SSI. Their formalised model of an SSI contains the definition of an entity. An entity has an identity which consists of of the union of all its partial identities. These partial identities are all of his attributes and values in a specific domain. Hence, an entity can be contained in multiple domains, where each partial identity can be subdivided into profiles (subsets of the attributes contained in the partial identity within a domain).

Cameron (2005)

Cameron (2005) describes one of the inherent flaws of the Internet being the lack of an identity layer: there is no standardised mechanism for identification, resulting in a shattered "patchwork of identity one-offs", so-called workarounds for identification.Cameron proposes a *unifying identity metasystem*, which, similarly to what sockets provide for networking, provides an abstraction for identification which allows application to abstain themselves from specific implementations and allow (lose) coupling of digital identities. For this, Cameron developed the seven *Laws of Identity*. These will be discussed more thoroughly in section .

Allen (2016)

Allen (2016) discusses the ten principles of SSI. Firstly, their work explains issues with traditional (physical) identity measures, e.g. driver licenses and social security cards, which are erroneously portrayed as identities. As a consequence, the issuing authority has the capability to nullify ones “identity".Allen propose SSI as an improvement and solution. Next, the four phases of evolution of identity are explained.

?

? present EverSSDI: a framework based on Ethereum smart contracts allowing for unique identifiers to normalise different user identities. Additionally, they construct an authorisation method based on Hierarchical Deterministic (HD) keys, an information verification mechanism and two methods for identity recovery. Their design makes use of Ethereum smart contracts to store encrypted fingerprint variants of claims. The design uses so-called “Ever-Service" servers to generate unique IDs named “Ever-IDs". These specific servers also aid in a login procedure. It is not clear who manages the “Ever-Service” servers. They introduce two methods for identity recovery: one based on SNS authorisation and one based on Ethereum Oracles. The authors mentioned that their future research will incoroporate a custom public blockchain.

Belchior et al. (n.d.)

Belchior et al. (n.d.) propose their Self-Sovereign Identity Based Access Control (SSIBAC) model: and SSI access control scheme based on blockchain technology. Their research contributions include an access control scheme based

on SSI, an implementation and evaluation. They achieve a throughput of 0.9 seconds per access control request. The design works by creating a verifiable presentation (VP) from a verifiable claim (VC). This VP is sent to a verifier, which confirms that the client holds the VC by verifying whether it satisfies a specific predicate. The drawback to the scheme is that the verifiers are a single point of failure in their design, which is acknowledged by the authors.

Solution

Hybrid-Revocation Model

A relatively unresolved aspect of Self-Sovereign Identity, is the ability to revoke previously signed claims. A rather trivial approach is to construct a central storage location in which anyone can store their revoked signatures. This has the drawback of introducing a central authority, which can be said to defeat the purpose of SSI. A central “banlist" authority would be a single point of failure and has the ability to be misused (e.g. allow for arbitrary censorship of claims).

Current approaches require a large degree of interactivity between the signee and verifier. In existing distributed approaches, a verifier suspecting a claim to be invalid must actively query the signer for validating whether the presented signature is not revoked. This has the drawbacks of requiring both parties (i.e., the verifier and the signee) to be online and requires a high throughput of transaction, as otherwise this check introduces large latency in the verification process. This process has been visualised in Figure 3, in which it can be seen that a claim is verified with the signee. This design is prone to variations, e.g. requesting a list of all revoked signatures. It can be noted that in case verification is required for each presented claim, signatures would intrinsically longer be required, as we can now simply verify with the signee whether the claim is valid.

In order to address the previously identified weak-points and shortcomings, we introduce a hybrid solution. This model aims to require no interactivity between a verifying party and a signing party and allows for offline verification. The schematic design is visible in Figure 4. The scheme builds upon our previously defined notion of Trusted Entities: each client aims to accept signatures signed by a trusted entity, hence, each client trusts any revocation made by said trusted entities. The HRM design uses a so-called Offline Revocation List (ORL), which comprises entries of revoked signatures from TEs. The ORLs are stored distributed across all clients and, hence, only contain revoked signatures from client which they trust. The ORL requires periodical syncing in order to stay up-to-date.

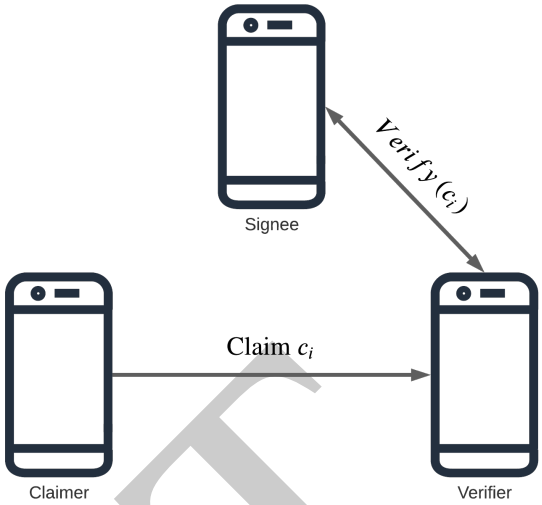

Figure 1

Revocation requiring interactivity

Revocation

Contributions

The work set out by Stokkink and Pouwelse and Stokkink, Epema, and Pouwelse will serve as a foundation of the IG-SSI scheme. The contributions made by this thesis will be an SSI scheme that can be said to be of *industry-strength*, which will be substantiated with a real-life trial of an implementation of said scheme. The main knowledge gap currently existing in the research area of SSI is the gap between the theoretical frameworks and the feasibility of these theories. E.g., strict processing latency requirements on mobile devices, communication overhead, and fault-tolerance. As such, this thesis will attempt to bridge this gap by constructing an SSI scheme together with developing an interaction model that allows for a practical implementation that is to be verified through real-life user tests.

Research Questions

The topic of Self-Sovereign Identity and the notion of *Industry-Grade Self-Sovereign Identity* shall foremost be investigated through the following research question:

“How can Self-Sovereign Identity be designed ”

This research question will allow for the investigation into and the development of a state-of-the-art SSI architecture. Based on the identified knowledge gap, the following sub-questions can be investigated:

1. *How to store verifiable claims locally in a decentralised fashion?*

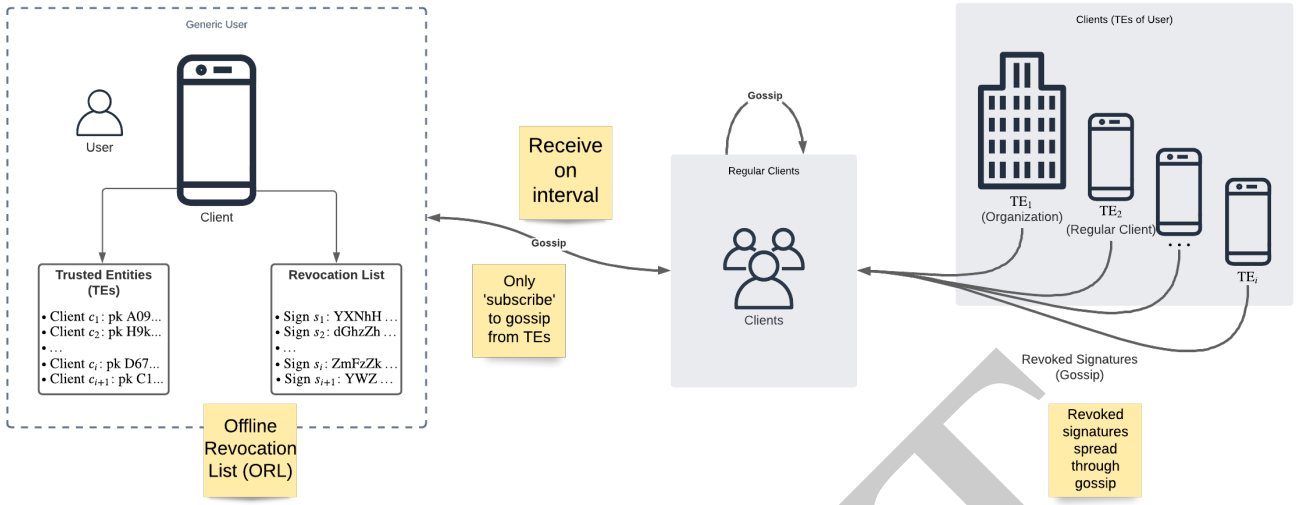

Figure 2

Hybrid-Revocation Model (HRM)

2. *How to integrate the concept of 'trusted entities' into Self-Sovereign Identity?*
3. *How to perform revocation without interactivity?*
4. *How to design an open Self-Sovereign Identity standard that allows for an accessible implementation (e.g. supported by all major smartphone operating systems?)*

Based on these results, we will be able to design an SSI architecture that will overcome these shortcomings and be deemed to be of *industry-strength*.

Solution

Hybrid-Revocation Model

A relatively unresolved aspect of Self-Sovereign Identity, is the ability to revoke previously signed claims. Whilst not necessarily being an issue solely present in SSI, distributed revocation is a rather unsolved issue. With distributed revocation, we speak about the notion of revoking signatures in a distributed fashion. Moreover, we append the additional requirements of non-interactivity and, as a consequence, offline usable revocation. In other words, revocation should not be dependent on (centralised) authorities, as this can have additional consequences on confidentiality and availability. As described by Khovratovich and Law (n.d.), the usage of authorities with revocation proofs, can lead to collusion. Therefore, relying on authorities for revocation can lead to the deterioration of privacy. More drastically, introducing authorities in revocation can lead to censorship, as these specialised nodes have the ability to either hide revoked signatures or to maliciously state signatures as being revoked. Hence, in order to address the additional raised issues, we present a truly distributed revocation mechanism.

Trivial Approaches

Revocation in general can be solved quite trivially. The first approach relies on the introduction of centralised authorities, the second approach requires the usage of distributed ledges, whilst the third relies on interactivity. These approaches can all utilise existing revocation mechanisms, designed for more closed identity ecosystems. For instance, the usage of backward unlinkable revocation described by Verheul (2016); the usage of revocable group signatures describe by Nakanishi, Fujii, Hira, and Funabiki (n.d.); or the usage of accumulators as described by Camenisch, Kohlweiss, and Soriente (n.d.); Camenisch and Lysyanskaya (2002).

Authorities

A rather trivial approach is to construct a central storage location in which anyone can store their revoked signatures. This has the drawback of introducing a central authority, which can be said to defeat the purpose of SSI. A central “banlist" authority would be a single point of failure and has the ability to be misused. Apart from availability issues, a single authority introduces a steep inequality across the network, as this client would have the ability to arbitrarily withhold revocations or may falsely introduce new ones. This effect may be counteracted by introducing several revocation nodes, e.g. per Sovrin’s design. However, this still leads to the requirement of interactivity, as communication with revocation nodes is still required for validation. Hence, we deem this trivial solution not sufficient for a truly distributed SSI system.

Distributed Ledgers

The usage of distributed storage solutions may appear to be quite suitable. The properties introduced by the usage of e.g. blockchain technology, can prove to build a resilient revocation mechanism. For instance, Lasla, Younis, Znaidi, and Ben Arbia (2018) describe a certificate revocation mechanism, tailored to Cooperative Intelligent Transportation Systems, utilising Blockchain technology. However, the introduction of distributed ledger technology, often imposes the issue of consensus. Requiring consensus algorithms such as Proof of Work or Proof of Stake, where the former introduces unnecessary power consumption, raising the entry barrier for IoT and portable devices. Apart from this drawback, offline validation of past blockchain transactions often require the storage of the entire chain. Where the most prominent blockchains, Bitcoin and Ethereum, require more than 300GB[10] and more than 200GB[11] for regular and 4TB[12] for archive nodes. Hence, offline validation would become quite infeasible for regular devices. Furthermore, requiring the communication with fully synchronised blockchain nodes, would replace transform the problem of interactivity within the SSI ecosystem, to one within the blockhain ecosystem, hence simply moving the problem instead of solving it. This makes the use of distribute ledgers not feasible for the imposed requirements.

Interactivity

The most trivial of solution may be to simply validate a credential by querying the authority of a credential. However, the imposes several restrictions on the validation process. Firstly, this requires the signee of the credential to be online. Availability in distributed systems is never a guarantee, hence, this introduces a weakness in the revocation mechanism. Secondly, interactivity with the signee removes any offline usability. As now, a connection to both the presenter and the signee must be made or the presenter must simultaneously make a connection to the signee in order to generate a non-revocation proof to present to the verifier. This make this approach not suitable.

The trivial solution all add a degree of interactivity or impose too strict of processing requirement to clients. Hence, the trivial solutions introduce requirements directly contradicting the properties sought after in the revocation mechanism. Hence, the aforementioned solutions are not suitable to solve the issue of revocation.

Remove?

remove

Current approaches require a large degree of interactivity between the signee and verifier. In existing distributed

[10]For Bitcoin blockchain size, see: `https://www.statista.com/statistics/647523/worldwide-bitcoin-blockchain-size/`

[11]For Ethereum blockchain size, see: `https://blockchair.com/ethereum/charts/blockchain-size`

[12]For Ethereum archive blockchain size, see: `https://etherscan.io/chartsync/chainarchive`

approaches, a verifier suspecting a claim to be invalid must actively query the signer for validating whether the presented signature is not revoked. This has the drawbacks of requiring both parties (i.e., the verifier and the signee) to be online and requires a high throughput of transaction, as otherwise this check introduces large latency in the verification process. This process has been visualised in Figure 3, in which it can be seen that a claim is verified with the signee. This design is prone to variations, e.g. requesting a list of all revoked signatures. It can be noted that in case verification is required for each presented claim, signatures would intrinsically longer be required, as we can now simply verify with the signee whether the claim is valid.

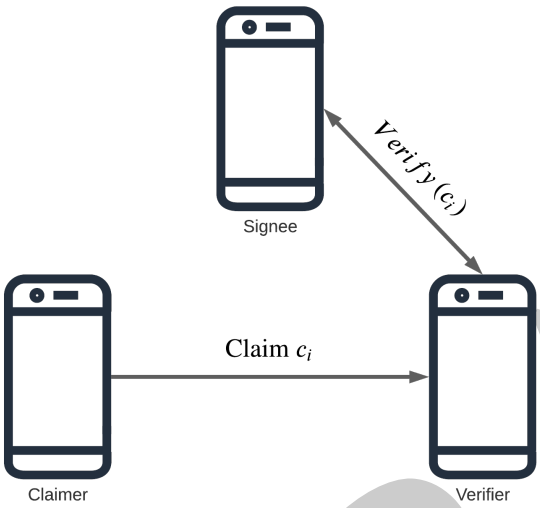

Figure 3

Revocation requiring interactivity

Design

In order to address the previously identified weak-points and shortcomings, we introduce a hybrid solution. This model aims to require no interactivity between a verifying party and a signing party during verification and allows for offline validation. The schematic design is visible in Figure 4. The scheme builds upon our previously defined notion of Trusted Entities: each client aims to accept signatures signed by a trusted entity, hence, each client trusts any revocation made by said trusted entity. The HRM design uses a so-called Offline Revocation List (ORL), which comprises entries of revoked signatures from TEs. The ORLs are stored distributed across all clients and, hence, only contain revoked signatures from client which they trust. The ORL requires periodical syncing in order to stay up-to-date.

Synchronisation

Synchronisation in the system is dependent on the entire community of peers. Whilst consensus on revoked signatures is reached on peer-level, propagation is dependent on the entirety of peers. I.e., revocations are sent across the network in a peer-to-peer fashion. More specifically, peers are to actively propagate the latest revocations to other peers by means of gossip. Gossip protocols are modelled after epidemic spreads. Similarly to how gossip can spread throughout an office building, epidemics spread viruses across hosts. Translated to distributed systems, clients attempt to spread the latest information to as much other clients as possible. The effects of this, is that information ripples through the entire network. As with epidemics and gossip, this ripple takes time to reach all peers. This time we refer to as the *propagation time*. Propagation time is dependent on multiple factors, both digital and physical.

The affecting factors of the propagation time can be split up into two factors: (1) the protocol characteristics (2) network properties.

Protocol Properties

For protocol delays, the propagation time is dependent on the parameters imposed on the protocol. The parameters related to peer-contacting directly impact the frequency of the gossip. These are:

1. **Gossip-interval** (t_g): the time interval on which peers are gossiped to.
2. **Gossip amount** (n_g): the number of peers which are gossiped to on a time interval.
3. **Peer selection** ($\mathcal{F}_g(\mathcal{X})$): the function used to determine which peers are gossiped to.

The reasoning that the throughput of gossip can be limited are due to client restrictions. A client can impose certain restrictions regarding the frequency of gossiping to peers. This can, for instance, be due to hardware restrictions or energy consumption limitations. The gossip-interval, amount, and peer selection process, directly influence the number of peers gossiped to clients per time interval, thus, directly impacting the propagation time. The delay presented by these parameters can be summarised to the following formula:

Let $P = \{p_0, \ldots, p_{n-1}\}$ be the set of peers of size n_p in the network and let $g = t_g \cdot \frac{n_p}{n_g}$ be the minimal number of interval iterations required to gossip to all peers. The peer selection function $\mathcal{F}_g(X)$ may result in overlapping subsets. I.e., let $f_i = F_g(P)$ be the subset of peers generated at iteration i and let $f_{i+j} = F_g(P)$ be the subset generated at iteration $i+j$, then it does not necessarily hold that $f_i \cap f_{i+j} = \emptyset$. Hence, let $P_f = p_0, ..., p_{n-1}$ be the multiset of peers of size $m_p >= n_p$

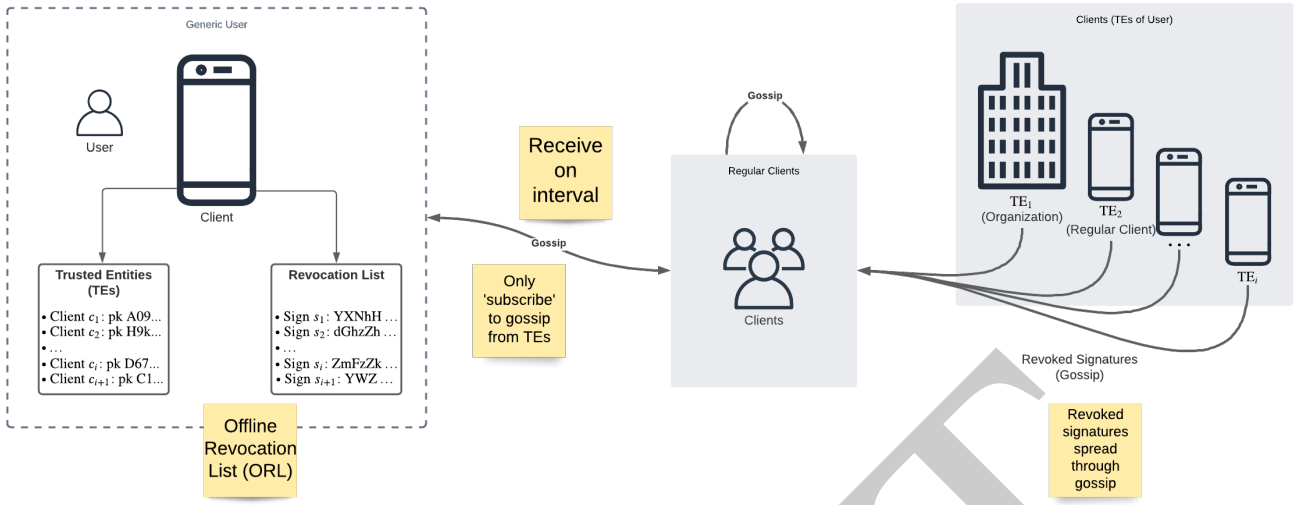

Figure 4

Hybrid-Revocation Model (HRM)

selected throughout each iteration until convergence. I.e., the peer selection function $\mathcal{F}_g(X)$ selected at least $m_p >= n_p$ peers, leading to at least $t_g \cdot \frac{m_p}{n_g}$ iterations. The additional iterations can be modelled by: $h = t_g \cdot \frac{m_p - n_p}{n_g}$, where $h \geq g$ This leads to the propagation time for the protocol delays for a single client i attempting to gossip a single update to the entire visible network with size n as to be as summarised in Equation 1.

$$\begin{aligned} \mathcal{T}_{protocol,i} &= h + g \\ &= t_g \cdot \frac{n_p}{n_g} + t_g \cdot \frac{m_p - n_p}{n_g} \\ &= t_g \cdot \left(\frac{n_p}{n_g} + \frac{m_p - n_p}{n_g} \right) \\ &= t_g \cdot \frac{m_p}{n_g} \end{aligned} \tag{1}$$

As clients are not aware of their position in the network (relatively to others) or of the peers already contacted by other clients, there can only be set an upper bound on the expected runtime of the algorithm, as each peer attempts to gossip all information to all other peers. Hence, we can summarise the propagation delay to the formula presented in Equation 2, where $t_{g,i}, m_{p,i}, n_{g,i}$ are the gossip-interval, number of selected peers, and gossip amount for client i, respectively.

$$\begin{aligned} \mathcal{T}_{protocol} &\leq \sum_{i=0}^{n-1} \mathcal{T}_{protocol,i} \\ &\leq \sum_{i=0}^{n-1} \left(t_{g,i} \cdot \frac{m_{p,i}}{n_{g,i}} \right) \end{aligned} \tag{2}$$

Due to parameters being dependent on hardware and deployment restrictions, there does not exist an optimal setting for all deployments types. Depending on the expected frequency of updated data, different parameters may be suitable. Different configurations lead to different characteristics imposed on the system. Increasing the gossip-interval leads to, generally, more up-to-date peers as a client will gossip the latest information more frequently. Whilst increasing the amount of gossip will allow for more clients to receive information, whilst not necessarily leading to more up-to-date clients. Where up-to-date refers to possessing the latest information. This is, of course, dependent on the frequency of new information. The peer selection function can influence the number of up-to-date and the number of updating clients both positively and negatively, as the peer selection function $\mathcal{F}$ allows for multiple modus operandi. E.g., the $\mathcal{F}$ can be a pseudo-random function (PRF), in which the peers are selected arbitrarily, giving each subset of clients of size n a near equal chance of being gossiped to on each interval $\mathcal{T}$. However, such an approach may lead to specific peers being selected multiple times, due to chance, at an interval. Hence, possibly negatively impacting the overall propagation time. A more sophisticated is also possible: e.g. a combination

of a PRF with backtracking, in which a subset is dropped in case a member of the set has been contacted in the last m iterations. Such an approach can prove to increase the overall throughput of information, thus decreasing the propagation time.

These three parameters do not necessarily have to be static: clients can record the latest gossip sent to specific peers, hence, selectively gossiping on new information. This can be extended to decreasing the gossip-interval and amount depending on the frequency of new information. This dynamic behaviour allows for more efficient usage of resources and decreases the overhead of gossiping to peers which may already have received the latest information. However, this would increase memory usages and runtimes, as now such metadata on gossiped information must be recorded by the client.

Network Properties

Foremost, the propagation time is dependent on the amount of nodes in the system. Where a system with a single node converges in a constant time. I.e., the system converges in c time with a system of size $n = 1$ nodes. For any larger sizes ($n > 1$), several constraints on the propagation time are introduced. Firstly, the size of the information itself becomes a factor: as the throughput of data between nodes may not necessarily be equal, the time for propagation between nodes may differ. More specifically, the propagation of information in a (sub)graph with $n > 2$ with a gossiping node n_i and two uninformed directly linked nodes n_j and n_k, may result in node n_k becoming informed prior to node n_j or vice versa. Reasonings for this are the imperfections present in the network infrastructure and deployment environment differences. For instance, network congestion present in the link to a certain node can lead to queueing delays and packet loss. Lower available bandwidth may also conceive such discrepancies. Differences in deployment environments (i.e., different hardware), may also lead to different convergence timings. For instance, a faster CPU and more available memory may lead to faster processing of gossip and, thus, a faster propagation time compared to weaker hardware. Hence, each node p_i introduces a relatively unique processing delay c_i. This processing delay will be constant for a single update iteration, i.e., this delay is initiated after another client gossiped new information to this client. However, this delay may differ on subsequent gossip, as this constant is influenced by factors such as the current load of the node and the size of the gossiped data. Therefore, we assume that this delay is of arbitrarily length, which only becomes apparent after a node has gossiped new information to this node. Hence, no prior analysis can be made with regard to this delay, we simply acknowledge its existence and, thus, base the network propagation delay on the minimum link with a gossiping node.

Next, we generalise the delays imposed by the network. Let $\delta_{i,j}$ be the propagation delay from node i to node j and let function $\Delta(p_j)$ be the smallest propagation delay for node p_j to be gossiped to. I.e., $\forall(p_i, p_k) \in \{p_0, ..., p_{n-1}\}$ it holds that $\delta_{i,j} < \delta_{k,j}$. Let $\mathcal{D} = \{\delta(p_0), \ldots, \delta(p_{n-1}\}$ be the set containing all these smallest propagation delays for each node. Finally, let $C = \{c_0, \ldots, c_{n-1}$ be the the set of delays imposed by processing times on the clients on invocation $\Delta(p_j)$. This leads to the network delay for a single client i updating the entirety of the to him visible networks with size n as summarised in Equation 3

$$\mathcal{T}_{network,i} = \sum_{j=0}^{n-1} \left(\delta_{i,j} + c_j\right) \tag{3}$$

The the total propagation time in a system with a set of $P = \{p_0, \ldots, p_{n-1}\}$ nodes of size n can be modelled as visible in Equation 6.

$$\mathcal{T}_{network} = \sum_{i=0}^{n-1} \left(\Delta(p_i) + c_i\right) \tag{4}$$

Finally, we can model the entire propagation time of a single node and the entire graph. The propagation time for a single node can be seen in **??**

$$\begin{aligned} \mathcal{T}_{tot,i} &= \mathcal{T}_{protocol,i} + \mathcal{T}_{network,i} \\ &= \left(t_g \cdot \frac{m_p}{n_g}\right) + \left(\sum_{j=0}^{n-1} \left(\delta_{i,j} + c_j\right)\right) \end{aligned} \tag{5}$$

The propagation time for a network of size n, is visible in **??**

$$\begin{aligned} \mathcal{T}_{tot} &= \mathcal{T}_{protocol} + \mathcal{T}_{network} \\ &\leq \left(\sum_{i=0}^{n-1} \left(t_{g,i} \cdot \frac{m_{p,i}}{n_{g,i}}\right)\right) + \left(\sum_{i=0}^{n-1} \Delta(p_i) + c_i\right) \\ &\leq \sum_{i=0}^{n-1} \left(t_{g,i} \cdot \frac{m_{p,i}}{n_{g,i}} + \Delta(p_i) + c_i\right) \end{aligned} \tag{6}$$

Revocation

References

Allen, C. (2016, 5). *The Path to Self-Sovereign Identity.* CoinDesk. Retrieved from https://www.coindesk.com/path-self-sovereign-identity

Belchior, R., Putz, B., Pernul, G., Correia, M., Vasconcelos, A., & Guerreiro, S. (n.d.). *SSIBAC: Self-Sovereign Identity Based Access Control* (Tech. Rep.). Retrieved from https://vonx.io/

Camenisch, J., Kohlweiss, M., & Soriente, C. (n.d.). *An Accumulator Based on Bilinear Maps and Efficient Revocation for Anonymous Credentials* (Tech. Rep.).

Camenisch, J., & Lysyanskaya, A. (2002). *Dynamic Accumulators and Application to Efficient Revocation of Anonymous Credentials* (Tech. Rep.).

Cameron, K. (2005). The laws of identity. *Microsoft Corp*, *5*, 8–11.

Decentralized Identifiers (DIDs) v1.0. (n.d.). Retrieved from https://www.w3.org/TR/did-core/

Der, U., Jähnichen, S., & Sürmeli, J. (2017). Self-sovereign identity - opportunities and challenges for the digital revolution. *arXiv preprint arXiv:1712.01767*.

Ferdous, M. S., Chowdhury, F., & Alassafi, M. O. (2019). In search of self-sovereign identity leveraging blockchain technology. *IEEE Access*, *7*, 103059–103079.

Khovratovich, D., & Law, J. (n.d.). *Sovrin: digital identities in the blockchain era* (Tech. Rep.). Retrieved from http://www.credentica.com/the

Lasla, N., Younis, M., Znaidi, W., & Ben Arbia, D. (2018, 3). Efficient Distributed Admission and Revocation Using Blockchain for Cooperative ITS. In *2018 9th ifip international conference on new technologies, mobility and security, ntms 2018 - proceedings* (Vol. 2018-January, pp. 1–5). Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/NTMS.2018.8328734

Lundkvist, C., Heck, R., Torstensson, J., Mitton, Z., & Sena, M. (n.d.). *UPORT: A PLATFORM FOR SELF-SOVEREIGN IDENTITY* (Tech. Rep.).

Mühle, A., Grüner, A., Gayvoronskaya, T., & Meinel, C. (2018, 11). *A survey on essential components of a self-sovereign identity* (Vol. 30). Elsevier Ireland Ltd. doi: 10.1016/j.cosrev.2018.10.002

Nakanishi, T., Fujii, H., Hira, Y., & Funabiki, N. (n.d.). *Revocable Group Signature Schemes with Constant Costs for Signing and Verifying* (Tech. Rep.).

Othman, A., & Callahan, J. (2018, 10). The Horcrux Protocol: A Method for Decentralized Biometric-based Self-sovereign Identity. In *Proceedings of the international joint conference on neural networks* (Vol. 2018-July). Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/IJCNN.2018.8489316

Reed, D., Law, J., & Hardman, D. (2016). *The Technical Foundations of Sovrin A White Paper from the Sovrin Foundation* (Tech. Rep.).

Sovrin ™ : A Protocol and Token for Self-Sovereign Identity and Decentralized Trust A White Paper from the Sovrin Foundation (Tech. Rep.). (2018).

Stokkink, Q., Epema, D., & Pouwelse, J. (2020). A Truly Self-Sovereign Identity System. *arXiv preprint arXiv:2007.00415*.

Stokkink, Q., & Pouwelse, J. (2018). Deployment of a blockchain-based self-sovereign identity. In *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)* (pp. 1336–1342).

Tobin, A., & Reed, D. (2016). The inevitable rise of self-sovereign identity. *The Sovrin Foundation*, *29*(2016).

Verheul, E. R. (2016). *Practical backward unlinkable revocation in FIDO, German e-ID, Idemix and U-Prove* (Tech. Rep.).

Zimmermann, P. (1999). *Why I Wrote PGP.* Retrieved from https://www.philzimmermann.com/EN/essays/WhyIWrotePGP.html