ROWDY MITCHELL CHOTKAN

Industry-Grade Self-Sovereign Identity

Delft University of Technology

Ministry of the Interior and Kingdom Relations

To obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Monday August 30, 2021 at 10:00 AM.

Industry-Grade Self-Sovereign Identity

On the Realisation of a Fully Distributed Self-Sovereign Identity Framework

[DRAFT]

Author
R.M. Chotkan

Supervisors
Dr. J.A. Pouwelse, TU Delft
A. De Kok, RvIG

Student number:	4570243	
Project duration:	November 9, 2020 – August 30, 2021	
Thesis committee:	Dr. J.A. Pouwelse,	TU Delft, supervisor
	Dr. E. L. Brown,	TU Delft
	Ir. A. Aaronson,	Acme Corporation

Abstract

Preface

Contents

Abstract iii

Preface v

I Article 1

II Supplementary Material 13

A Background Information 15
A.1 Identity . . . 15
A.2 Digital Identity . . . 15
A.3 The Evolutions of DIMS . . . 16
A.4 Shortcomings in the Current Ecosystem . . . 18
A.4.1 Problems for Identity Providers . . . 18
A.4.2 Problems for Users . . . 18
A.5 Theoretical SSI Models . . . 19
A.5.1 The Laws of Identity . . . 19
A.5.2 The Path to Self-Sovereign Identity . . . 20
A.5.3 The Pyramid of Sovereignty . . . 21
A.6 Existing Solutions . . . 22
A.6.1 History of Self-Soveregn Identity . . . 22
A.6.2 Critique of the Term . . . 23
A.6.3 Sovrin . . . 25
A.6.4 Serto (uPort) . . . 25
A.6.5 Decentralized Identifiers . . . 25
A.7 Related Works . . . 25
A.7.1 Mühle et al., 2018 . . . 26
A.7.2 Der et al., 2017 . . . 26
A.7.3 Stokkink and Pouwelse, 2018 . . . 27
A.7.4 Othman and Callahan, 2018 . . . 27
A.7.5 Ferdous et al., 2019 . . . 27
A.7.6 Zhou et al., 2019 . . . 27
A.7.7 Belchior et al., 2020 . . . 28
A.8 Ethical Implications . . . 29
A.8.1 The End of Privacy . . . 29
A.8.2 Yet-Another-Authentication-System . . . 29

B Design 31
B.1 Preliminaries . . . 31
B.1.1 Attestations . . . 31
B.2 Architecture . . . 34
B.2.1 Required Functionalities . . . 34
B.3 Attestation Signing . . . 35
B.3.1 Attestation Flow . . . 36
B.4 Attestation Presentation . . . 38
B.5 Attestation Verification . . . 39
B.6 Attestation Revocation . . . 40
B.6.1 Remove? . . . 41
B.7 Design . . . 41

C Analysis & Results 45
C.1 Results 45
C.2 Runtime Analysis 45
C.2.1 Synchronisation Upperbound 45

List of Figures

A.1 Identity Groups 16
A.2 Balance of Power 19
A.3 The Pyramid of Sovereignty 22

B.1 The relationship between credentials and (verifiable) claims 31
B.2 Attestation Flow 35
B.3 Token Chain 36
B.4 Data Structure Relationships 37
B.5 Attestation Presentation 38
B.6 Interactive Verification 39
B.7 Revocation requiring interactivity 42
B.8 Hybrid-Revocation Model (HRM) 42
B.9 Multi-step Update Procedure 43

C.1 UDP Impact ($n = 1, m = 1$) 45
C.2 UDP Impact ($n = 1, m = 10$) 46

List of Tables

A.1 The principles by Loffreto (2016) 23
A.2 The properties defined by DID 26

B.1 Verifiable Revocation Update Format 44

Acronyms

DIMS Digital Identity Management System. x, 19, 20

HRM Hybrid-Revocation Model. x

IG-SSI Industry-Grade Self-Sovereign Identity. x, 31

KYC Know Your Customer/Client. x

PII personally identifiable information. x, 18, 19

SSI Self-Sovereign Identity. x

SSS Shamir's Secret Sharing. x

Glossary

Attestation In terms of SSI, an Attestation is a declaration made by an Authority to vouch for the validity of a Claim. x

Authority In terms of SSI, an Authority is any party attesting to a claim made by a Subject. x

Claim In terms of SSI, a Claim is a certain piece of information referring to a Subject, who is also the owner. x

Credential In terms of SSI, a Credential is a (set of) Verifiable Claim(s) which serve to authenticate data regarding a Subject. x

Subject In terms of SSI, a Subject is any party for which an Attestation is made or whom is the owner of a Credential. x

Verifiable Claim In terms of SSI, a Verifiable Claim is a claim of which its validity can be verified. x

Verifier In terms of SSI, a Verifier is any party verifying a Credential. x

I

Article

[DRAFT] Industry-Grade Self-Sovereign Identity

On the Realisation of a Fully Distributed Self-Sovereign Identity Framework

R.M. Chotkan and J.A. Pouwelse
R.M.Chotkan@student.tudelft.nl, J.A.Pouwelse@tudelft.nl

***Abstract*—The internet was created without a standardised identity layer, resulting in each user having to manage a plethora of digital identities which hold no legal value, often requiring cumbersome identity card checks, e.g., through digital photocopies. Initiatives such as *User-centring identities* have mostly failed, resulting in asymmetrical control over our digital data held by Big Tech. Self-Sovereign Identity (SSI) can prove to overcome these hurdles. SSI aims to put one at the centre of their digital presence, making them the owner of their identity. This enables full control over your own data and opens up the possibility of legally valid digital identities. We present Industry-Grade Self-Sovereign Identity (IG-SSI): a fully distributed SSI framework, requiring no specialised nodes or hardware, in which equality and offline usability are at the core of the design. The resulting schema allows for attestation signatures, presentation, verification, and revocation through Zero-Knowledge Proofs (ZKPs). Fully distributed revocation is achieved through the Hybrid Revocation Model (HRM): a gossip-based revocation model enabling offline verification. The HRM shows improvements with respect to already presented revocation designs and portrays great scalability. IG-SSI has been validated through field labs and has been designed in collaboration with the Dutch Ministry of the Interior and Kingdom Relations, we hope that SSI in general gains traction and, as such, a legally valid Self-Sovereign Identity may soon be deployed.**

I. Introduction

Since the dawn of the Information age, digital trust has been an issue requiring many workarounds. The core concepts of the internet are simply not built with trust in mind: there exists no standardised identity layer. As a result, the current landscape of identification and authentication mechanisms form a digital ecosystem of "digital one-offs" (Cameron, 2005). The popularity of identity management solutions by Big Tech has resulted in an oligopoly in digital identity Siftery (2017). Wherein a regular oligopoly consumers are at a disadvantage price-wise (Stigler, 1964), in this technical oligopoly the identity providers have an asymmetrical control of ones digital presence. Furthermore, increasing needs for digital identities from governments such as the European Union, has catapulted the research into and relevancy of the field itself. With the State of the Union 2020 address by President Von der Leyen portraying the relevancy of the problem:

"*Every time an App or website asks us to create a new digital identity or to easily log on via a big platform, we have no idea what happens to our data in reality. That is why the Commission will soon propose a secure European e-identity. One that we trust and that any citizen can use anywhere in Europe to do anything from paying your taxes to renting a bicycle. A technology where we can control ourselves what data and how data is used.*"

The need for digital identity, furthermore, stems from the urgency of COVID-19 vaccination passports, requiring digital verifiability and validity across borders European Commission (2021). This digital and societal gap can prove to be filled by the *Self-Sovereign Identity* (SSI) concept. SSI aims to generate digital trust by providing verifiable digital identities, putting the user at the centre. SSI is a concept requiring cutting-edge concepts such as decentralised ledger (DL) technology and decentralised public key infrastructure (DKPI). As such, the feasibility of developing a schema that is both technologically and usability-wise sound, can be proven to be difficult. Several solutions exists (e.g. Sovrin[1], Serto[2] and Irma[3]) many of which, however, require specialised infrastructure limiting equality in the network or do not stem from academia, making their results more difficult to reproduce and limiting the analysis of their design choices.

This article introduces *Industry-Grade Self-Sovereign Identity*: an academic Self-Sovereign Identity framework focusing on distributed revocation, offline verification, and intrinsic equality across the network. The scheme is based on the previous works by Stokkink & Pouwelse (2018); Stokkink et al. (2020). The following main contributions are made: (1) The Hybrid-Revocation Model (HRM): a novel distributed revocation algorithm and (2) offline verification of verifiable claims (VCs). Secondly, a reference implementation of the semantic layer is created using the IPv8 protocol stack (Halkes & Pouwelse, 2011; Zeilemaker et al., 2013) as well as a proof-of-concept application portraying feasibility on handheld devices.

II. Design

Self-Sovereign Identity is built around *Verifiable Claims* (VCs) (Mühle et al., 2018). VCs are composed of several terms as visible in Figure 1. Firstly, a *claim* is made by a Subject (Sporny et al., 2019). Authorities can attest to a claim, making it a VC. When metadata is added to a VC, we speak of an Attribute. Finally, a set of related attributes is referred to as a *Credential* (Mühle et al., 2018).

In IG-SSI, the aforementioned differentiations are stored as presented in Figure 3. Each *Claim* is represented by an anonymised *Token*, which stores a reference to a claim via its hash. A *Token* can be references by multiple *Metadata* structures which can, hence, assign different properties to a *Claim* (e.g. a validity term). Furthermore, multiple *Attestations*

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

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

[3] For Irma, see: `https://irma.app/?lang=en`

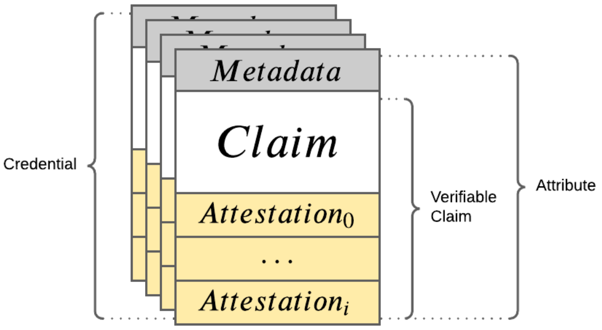


Fig. 1: Credential Structure

can be made for a *Metadata*. Finally, although not explicitly modelled, multiple *Credentials* can reference multiple *Attestations* and as such, multiple *Claims*. The *Tokens* are stored in a Blockchain-esque structure, referencing the previous Token as visualised in Figure 2. This aids in preventing the withholdment of claims as well as making it more difficult for one to use stolen credentials. The first token, comparable to a genesis-block in Blockchain structures such as Nakamoto (2009), contains the hash of the public key of the Subject. Any subsequent Credential, thus, generates a new Token, occupying a place as a shackle in the chain. As such, it is improbable for a client to attempt to hide the existence of an attestation or attempt to cheat the system, as otherwise the attestations of other Authorities become invalid (as the hash of the token will no longer be correct).

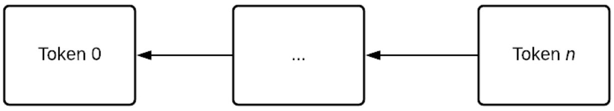


Fig. 2: Token Chain

Next, we discuss the lifecycle of these credentials. We identify four main interactions surrounding VC:

1) Attestation Signing
2) Attribute Presentation
3) Attribute Verification
4) Attestation Revocation

Furthermore we identify secondary issues regarding SSI **[TODO:** see if this should be added]:

A. Attestation Signing

The attestation procedure is visible in Figure 4. The design uses multiple phases and draws the distinction between two types of requests:

1) *Claim request*: Claims can be said to be the core data type. They are responsible for incorporating a specific value into a Zero-Knowledge Proof (ZKP). As visible in Figure 4, the design of claims allow for multiple *proof formats*. This disallows the lock-in of specific proof types, as any client can propose the usage of a specific proof, which can be used as long as the corresponding Authority supports the proposed type as well.
2) *Attestation request*: this type of request handles the actual attestation to claims, making them VCs. As such, an attestation always contains a reference to a claim. Multiple authorities can attest to the same claim. The attestation also refers to metadata, for instance containing validity terms and signature dates. This metadata, hence, creates an attribute. Note that credentials are not specifically defined. As they are merely a set of attributes they can be indirectly defined through the selection of multiple attributes.

The attestation flow consists of two phases, the Claim-phase and the Attestation-phase which do not necessarily require subsequent execution. More specifically, for a single to be attested claim, the Claim-phase requires a single execution, which must occur before the Attestation-phase. Whilst subsequently, the Attestation-phase can be performed indefinitely by different Authorities.

1) Claim-phase: The Claim-phase is initiated by Subject aiming to have a claim attested to by an Authority. During the initial request, the Subject makes the attribute name, the to be used proof format, his public key apparent, and any additional metadata apparent This public key belongs to a single use key pair, which aids in strengthening the privacy principle **[TODO:** see if this fits.]. The receiving Authority may respond to the request by generating a Claim incorporating the ZKP of the type defined by the *proof format*. This claim is sent back to the requesting Subject. After having received the Claim, the Subject moves on to the *Attestation-phase*. As may become apparent from this description two modus operandi are possible. Firstly, a client may self-create this claim, following the natural description of a claim. However, a client may not know the associated claim. Hence, the second modus operandi delegates the creation of the claim to an Authority, not requiring prior sharing of the claim value.

2) Attestation-phase: In the Attestation-phase, a requesting Subject requests an attestation for a Claim, creating a VC and subsequently an Attribute. When a Subject requests an attestation from Authority it also discloses the already created attestations as well as some prior Tokens. Furthermore, the Subject creates the *metadata* structure, allowing for properties for the attestation. The requested Authority can then verify the validity of the token chain as well as request verification of the previous claims. After which it can then generate a signature for the hash of the corresponding metadata, which serves as the actual attestation. Furthermore, this metadata incorporates the hashed value of the underlying plaintext value, which is discussed further during the *Verification* phase. However, as a hash would allow for trivial preimage attacks for attributes with a limited message space (e.g. an *age* credential), we propose the usage of salts Arias (2021).

B. Presentation Flow

The interactions for the presentation of attributes is portrayed in Figure 5. In this structure, an Authority requests an

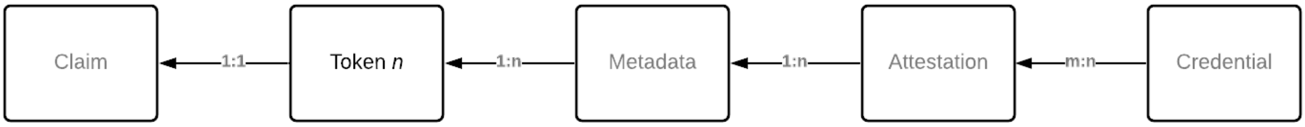


Fig. 3: Datastructures in IG-SSI

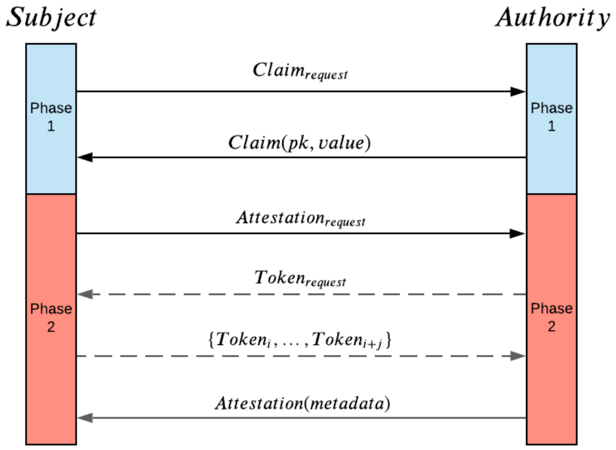


Fig. 4: Attestation Flow

attribute with a specific name. A Subject may subsequently decide whether to respond to such a request and to disclosure the corresponding attribute. Next, similarly to the Attestation flow, an Authority may request the tokens of previous claims until has gain enough confidence. Note here that the credential request is not necessarily required, as a client can disclosure an attribute directly. However, the specification of an attribute name aids in selective disclosure whilst additionally allowing the Authority to determine whether a specific credential is solicited. After a credential has been disclosed and, thus, presented, the Authority may verify its validity.

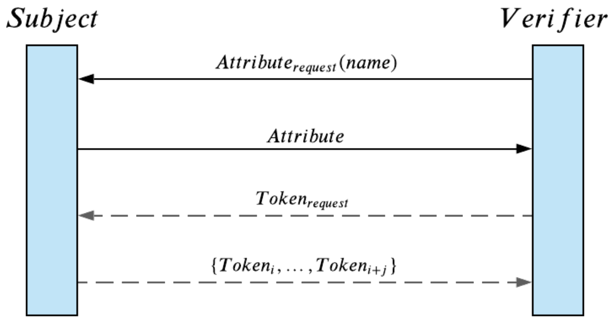

$Subject$
$Verifier$
$Attribute_{request}(name)$
$Attribute$
$Token_{request}$
$\{Token_i, \ldots, Token_{i+j}\}$

Fig. 5: Attestation Presentation

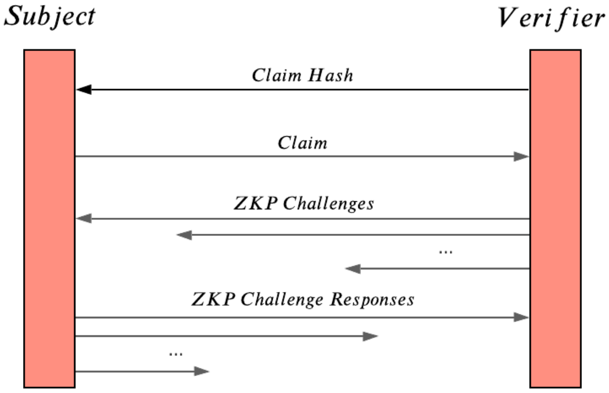


Fig. 6: Interactive Verification

C. *Verification Flow*

We propose two types of verification: an interactive a non-interactive variant. Both methodologies use the attestations made by authorities. Hence, the list of attestors must contain an Authority that is trusted by the Verifier.

The former variant is presented in Figure 6. For active verification, a Verifier requests the underlying Claim by presenting the claim hash to the Subject (procured through e.g. a Token). The Subject may consent by sending the requested Claim. Next the Verifier may send challenges to verify the underlying ZKP. Note that for this to happen, the Authority must either be already aware of the value belonging to the attribute or the plaintext value must be shared. Sharing of the plaintext value can be done during presentation-time. This should be performed using encryption in order to preserve privacy. Furthermore, the Authority verifies the presented attestations.

The second method for verification solely uses the attestations. In order for this attestation to pass, the list of attestors must contain an authority that is trusted by the Verifier. If this is the case, a Verifier may accept the value proposed by the Subject in case the metadata contains the hash of this value and the signature made by one of the acknowledged authorities over the metadata is valid. This approach does not require any connectivity between the Subject and Verifier, apart from the presentation itself. However a presentation does not necessarily require any form of digital communication (e.g. it can be performed through QR-codes), allowing full offline verification. It is, however, to note that this offline verification, thus, does not rely on any additional token requests and, as such, all tokens must either be made directly apparent to the

Verifier during presentation-time or the verifier must make its decision based on the presented Attestation and his reliance on and knowledge of acknowledged authorities.

D. Revocation

Specialised verification nodes for distributing and managing revocation present in Zhou et al. (2019); Tobin & Reed (2016); by Design Foundation (n.d.), would deteriorate the equality in the network and could even lead to censorship or collusion Khovratovich & Law (2017). As IG-SSI is designed without such nodes the trivial solution for revocation is to actively query the Authorities in order to verify that they still attest for a claim Stokkink & Pouwelse (2018). This querying requires interactivity with the Authorities. As a consequence, offline verification becomes impossible. Whilst availability often is a key characteristic in distributed systems, there is no guarantee that specific clients are available. As IG-SSI allows for an indefinite amount of attestations for a single Claim, interactivity with the Authorities can prove to become rather unmanageable as it introduces additional verification time due to additional network traffic and response times of Authorities.

1) Hybrid-Revocation Model: HRM attempts to overcome the hurdle of interactivity whilst allowing for offline-verification. The *hybrid* nature of the model stems for its offline capabilities: during verification-time, clients do not require to be online. They merely require occasional synchronisation of revoked attestations through communication with other peers. The general flow of the design can be seen in Figure 7. The protocol has three key concepts:

1) Trusted Authorities (TAs)
2) Propagation Algorithm
3) Offline Revocation List (ORL)

Next, we explain each concept.

Trusted Authorities

A criterion on which a client is able to determine the validity of a revocation is whether the revoking Authority is trusted by the client. This is where we introduce the notion of Trusted Authorities (TAs). As mirrored by real life, a person has (relatively speaking) a choice whether to acknowledge a certain authority. With SSI aiming to be a digital extension to one's identity, one should also be able to make such an acknowledgement in the digital domain.We propose the usage of a Trusted Authority Storage (TAS). In the TAS, the public key and the public key hash of a TA are stored. We make the distinction between acknowledged (trusted) and Unacknowledged Authorities (UAs). As discussed previously, client roles are neither static nor mutually exclusive. As a consequence, potentially every client can be an Authority. However, it is up to a client to determine whether an authority is a TA or an UA. In terms of distributed revocation: a client aims to accept only those revocations of TAs. The results of acceptance are the storage of the revoked signatures and propagation towards network.

Propagation

In order to safeguard availability in the network we propose the propagation of revocations through gossip. This requires a verifiable revocation format and a propagation protocol. We propose the structure as visible in Table I. This design, in addition to the revoked hashes, includes a public key hash, a version number, a specification for the used hashing algorithm and a signature. The public key hash allows for the retrieval of the public key in case said key belongs to the TAS of the receiving client. If this is the case, the signature can be verified by concatenating the version number with the revocations. Unique version numbers allow clients to determine whether certain revocations are missing from their ORL. The revocations themselves are the hashes of the Credential's Metadata. This invalidates any attestations made to this metadata and the token it points to. The specification of the hashing algorithm disallows implementation lock-in. algorithm 1 and algorithm 2 portray the algorithms for gossiping revocations and requesting updates, respectively.

The propagation itself requires a protocol that ensures information is (eventually) spread across the entire network, whilst also ensuring that unavailable nodes receive the information at a later instance. For this, we propose the usage of gossip protocols with interval re-transmission. Gossip protocols are communication protocols which allow for the periodic exchange of data with (random) peers (Kwiatkowska et al., 2008).

Furthermore, in order to decrease the overhead of gossiping a theoretically unbound number of signatures, we propose the usage of a multi-step update procedure. This procedure has been visualised in Figure 8. This procedure is split-up in two phases: firstly, a gossiping client gives notice to another client that it possesses specific authority-version pairs, containing the public key hash of an authority and the latest version it is aware of. Next, the receiving client can request an update by sending back the latest versions of the revocations stored in their TAS. This allows a client to selectively send updates, as the receiving party makes an underbound of the known versions apparent. This extra step of counteracts overhead as clients are not necessarily interested in specifics revocations as the authority may be considered an UA or a client may already be synced.

We note that this procedure may be fine-tuned through the usage of revocation dates. Revocation dates may allow clients to opt out of old revocation versions, optimising storage usage as old revocations may no longer be relevant in the system due to the validity terms of the attestations having passed.

Offline Revocation List

Any valid received revocation should be stored by a client for later reference in the Offline Revocation List (ORL). Whilst no specific storage structure is required, we do propose the usage of Bloom filters for member checking. A Bloom filter is a memory- and time-efficient probabilistic data structure, which allow for efficient membership operations (Bloom, 1970). Raya et al. (2007, 2006) discuss the benefits of Bloom filters in Certificate Revocation Lists (CRLs), which can be transformed to our concept of ORL, as the ORL can be deemed

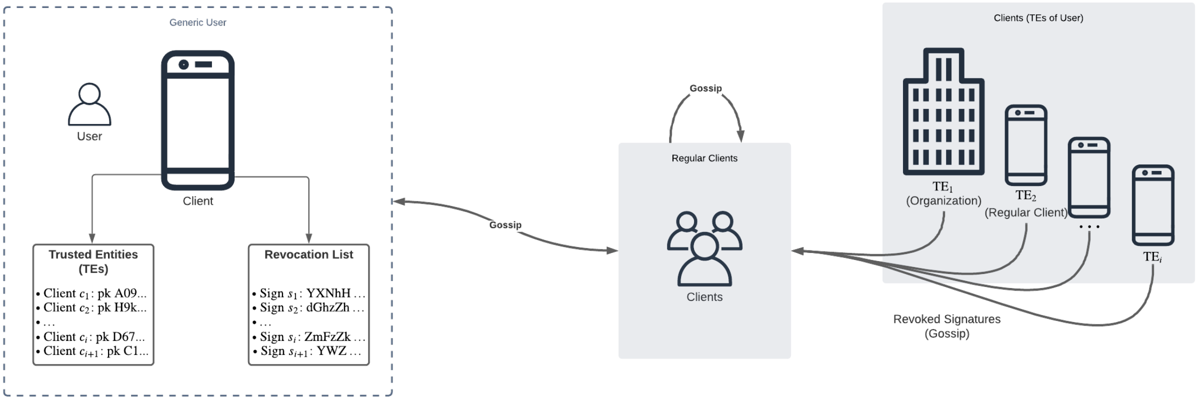


Fig. 7: The Hybrid Revocation Model (HRM)

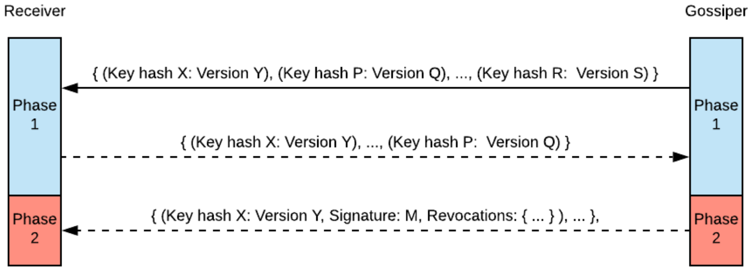

Receiver
Gossiper
Phase 1
Phase 1
Phase 2
Phase 2
{ (Key hash X: Version Y), (Key hash P: Version Q), ..., (Key hash R: Version S) }
{ (Key hash X: Version Y), ..., (Key hash P: Version Q) }
{ (Key hash X: Version Y, Signature: M, Revocations: { ... }), ... },

Fig. 8: Multi-step Update Procedure

a more generic variant of a CRL.

As yearly up to 400 thousand identity documents are stolen in the UK (HM Passport Office & The Rt Hon Caroline Nokes MP, 2018), this magnitude of revocations must be feasible. Revocation membership checking can prove to become quite expensive both memory- and runtime-wise. Even efficient algorithms such as Binary search (runtime complexity of $\mathcal{O}(\log(n))$ can lead to too high run time to be usability. As such, we propose the usage of Bloom filters for membership verification, in which a search on the actual data is only performed in case of a possible match. Bloom filters suffice as the probability of encountering a revoked attestation is expected to be low, as we assume the majority of the nodes to be honest as fraud can be detected.

Furthermore, we note that the ORL can be replaced by a Bloom filter entirely. A client may chose to accept the probabilistic nature of Bloom filters over the exact membership check from memory. Such nodes may not be able to aid in the propagation of the revocations, however, the low memory requirements may prove to make the protocol suitable for IoT devices.

TABLE I: Verifiable Revocation Update Format

Authority key hash	5e2bf57d3f40c4b6df6...
Version	1701
Hashing Algorithm	SHA3-256
Signature	422c06fbb4fbd23d33...
Revocations	b788c5b28dba2fc6a0... 7f2519609cf157d7e9... ... e2d7610dcb53724675...

```
Algorithm 1: Revocation Gossip
input : Set of Clients in the network
        $\mathcal{C} = \{c_0, \ldots, c_i\}$, Set of known
        Authority-Version pairs
        $\mathcal{A} = \{(a_0, v_j), \ldots, (a_j, v_k)\}$ Gossip interval
        $t_g$, Peer selection amount $n_g$
output: Revocation update gossip
while True do
    $\mathcal{C}_g \leftarrow$ SelectPeers ($\mathcal{C}, n_g$);
    foreach $c_i \in \mathcal{C}_g$ do
        GossipRevocations($c_i, \mathcal{A}$);
    Wait($t_g$);
```

E. Analysis

We computed a theoretical upperbound on the HRM protocol, as visible in Equation 1. This upperbound is dependent on delays imposed by the protocol and the network. 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. Let $\delta_{i,j}$ be the propagation delay from node i to node

Algorithm 2: Revocation Update Request Procedure

```
input  : Set of Authority-Version pairs
         $\mathcal{A} = \{(a_0, v_j), \ldots, (a_j, v_k)\}$, Set of trusted
         Authorities (TAS) $\mathcal{T} = \{t_0, \ldots, t_n\}$
output: Revocation update request

On reception of $\mathcal{A}$ by Client $c_i$;
for Authority $a_i$, Version $v_j$ in $\mathcal{A}$ do
    if $a_i \in \mathcal{T}$ then
        $v_{local} \leftarrow$ FindMissingVersion($a_i$);
        if $v_{local} < v_j$ then
            RequestUpdate($c_i$,$a_i$,$v_j$);
```

j then function $\Delta(p_j)$ computes 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 $\delta_{i,j} < \delta_{k,j}$. Finally, let c_i be the delays imposed by processing times on the client i

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

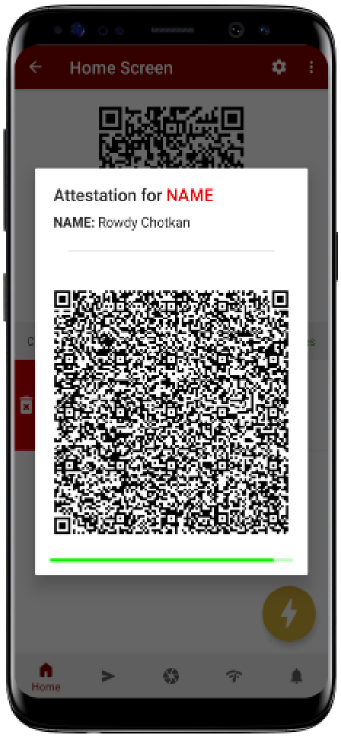
III. Implementation

section II presented a Self-Sovereign Identity framework based on the prior works by Stokkink & Pouwelse (2018); Stokkink et al. (2020) with the novel Hybrid-Revocation Model and offline verification capabilities. Based on this design, two implementations have been made using the IPv8 protocol stack [4]. The selection of IPv8 stems from firstly its academic background, proving its viability through various publications. Secondly, IPv8 allows for direct client-to-client communication, hence, enabling a fully distributed infrastructure at the core of the solution. Finally, IPv8 does not require (expensive) Proof-of-Work algorithms utilised by Blockchain structures such as Nakamoto (2009) and Buterin (2013).

Firstly, three semantic layers have been implemented on top of the Kotlin implementation of IPv8 [5]:

1) **Attestation Layer**: abstracts the creation and verification of Claims through Zero-Knowledge Proofs.
2) **Credential Layer**: abstracts the attestations of Authorities to Claims, hence creating Credentials, and enables chaining of attestations.
3) **Revocation Layer**: abstracts the handling of revocations of attestations.

Two types of ZKPs have been implemented: firstly, a ZKP proof allowing arbitrary data and the verification of exact values. The implementation is based on the algorithm proposed by Boneh et al. (2005), allowing verifiable computation through 2-DNF formulae over bits. Secondly, the range ZKP proposed by Peng & Bao (2010), allowing encoding of integer values laying in a specific range.The commitment scheme proposed by Boudot (2000) has been implemented in order to realise this range proof, based on the work by Stokkink & Pouwelse (2018). Both of these proofs are interactive. However, as shown by Koens et al. (2018), the schema introduced by Peng & Bao (2010) can be made non-interactive. The code for the reference implementation of these semantic layers is available on the IPv8 repository[6].



Attestation for NAME

NAME: Rowdy Chotkan

Fig. 9: SSI Demo Application

Secondly, a mobile client has been implemented in the form of an Android application. This client uses the implementation of the three semantic layers and showcases the usability on smart phones. The application supports all discussed communication per the three semantic layers. In addition, clients can create multi-party communication channels in order to force visibility with one another. This is performed through specialised tokens. The application enables offline verification through the presentation of Claims and attestation through QR-codes. As the Claims can comprise any form of data, the client even supports attestations to pictures; opening up the possibility for digitally attested to passport photographs. An example of the user interface is visible in Figure 9.

The implementation can be found on the Trustchain superapp repository[7].

[4] For the official (Python) documentation of IPv8, see: `https://py-ipv8.readthedocs.io/en/latest/`

[5] For the Kotlin implementation of IPv8, see: `https://github.com/Tribler/kotlin-ipv8`

[6] For the Kotlin IPv8 repository, see: `https://github.com/Tribler/kotlin-ipv8`

[7] For the Android application, see: `https://github.com/Tribler/trustchain-superapp`

IV. Analysis

A. Experimental Setup

We test the HRM protocol through emulation of IPv8 clients. The emulation was performed on a system with an i7-6700HQ and 16GB of RAM. As IPv8 uses UDP, packet loss introduced additional constraints on the testing setup when using multiple clients. In the performed measures this was counteracted through manual delays. As such, each presented measure contains an additional adjusted result. The systems allowed for up to 11 simultaneous clients. For revocations, we generated datasets of 32 bytes SHA3-256 hashes, which are used in the implementation of IG-SSI. Revocations were split-up into sets of 1000 in order to minimise the impact of a single packet loss. For the default parameters, the gossip-interval t_g was set to a minimum of $50ms$ in order to maximise throughput of gossip. The number of selected peers m_p was set to 5, as IPv8 recommends up to 30 simultaneous connections, such a smal amount suffices. Additionally, this number is of little impact due to the low gossip-interval. Finally, the gossip amount (n_g) used is mostly 1000, as this number led to fewest packet loss and the number of revocations scale linearly, as show in Figure 10a.

B. Revocation Amount

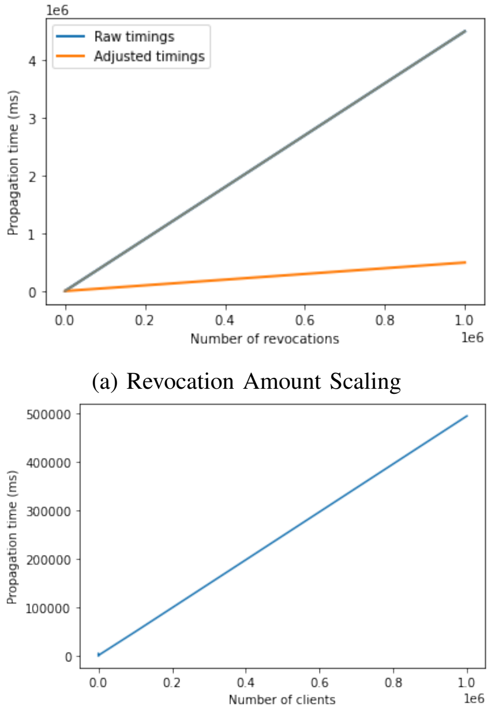
Figure 10a showcases the revocation scaling in a system of 1 Authority and 10 regular clients ($n = 1, m = 10$). As visible, the number of revocations scale linearly with the amount. In this setup, up to 1 million revocations were used, with increments of a factor 10. The adjusted rate showcases the performance adjusted for the manual delays. As visible in Figure 10, 1 million revocations take roughly 500 seconds or 8 minutes. As this can be deemed more than two years worth of revocations HM Passport Office & The Rt Hon Caroline Nokes MP (2018), we deem this scalability usable. Furthermore Figure 11a portray this scaling for a single client. As expected, this result scales linearly with the amount of revocations as well. With a single client taking roughly 10 seconds to update 10 thousand revocations. For 1 million, this would take roughly 3 minutes. As such, we note that the single Authority can be a limiting factor. This can due to system or implementation constrains. Finally Figure 11 portrays the scalability using TFTP in IPv8. We note that due to packet loss this method is not practical for our use case.

C. Client Scaling

Figure 12 portrays the effect of the number of clients on the propagation time. As visible the number of clients does not pose a large impact. It can be seen that the propagation time roughly doubles with the increase of 10 clients. As such, each additional client appears to a 10 percent propagation delay. **[TODO:** double check this experiment**]**. Indicating a linear increase.

D. Gossip Interval

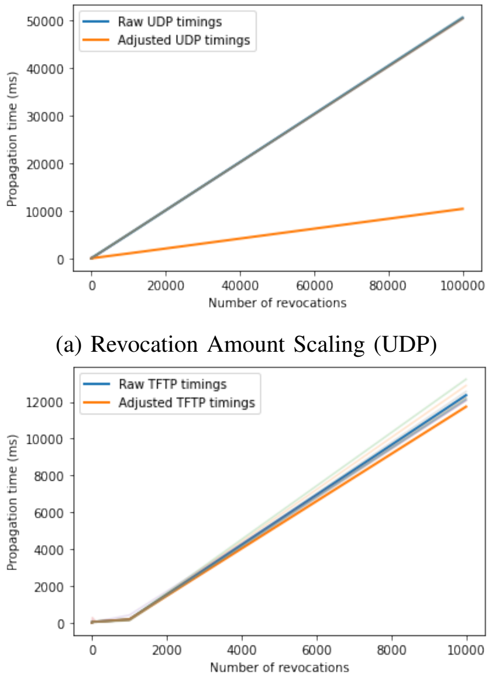
The impact of the gossip interval is visible in Figure 13. This parameter appears to have a linear effect as well.



(a) Revocation Amount Scaling

(b) Revocation Amount Scaling (adjusted measure only)

Fig. 10: Revocation Amount Scaling $(n = 1, m = 10)$



(a) Revocation Amount Scaling (UDP)

(b) Revocation Amount Scaling (TFTP)

Fig. 11: Revocation Amount Scaling $(n = 1, m = 1)$

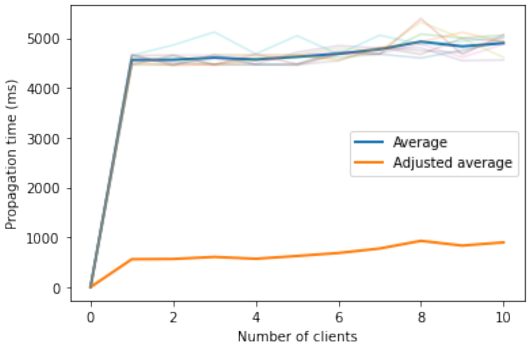


Fig. 12: Scaling of Clients

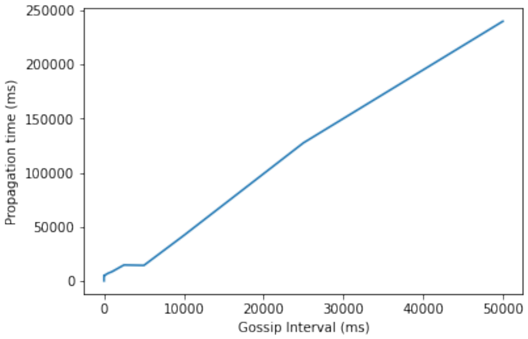


Fig. 13: Gossip Interval Scaling

E. Gossip Interval

[TODO: add]

F. Bloom filter

For the ORL, a Bloom Filter (Bloom, 1970) has been implemented for memory-usage and run-time improvements. Based on the expected 400.000 lost identification documents per year, as presented by HM Passport Office & The Rt Hon Caroline Nokes MP (2018), the following memory and time considerations can be made. Firstly [TODO: update], a storage for 300.000 hashes of 32 bytes each, results in a space usage of at least 9.2 megabytes. Whilst a Bloom filter with a probability of a false positive of 1 in 100 million and 27 hashing functions, can achieve such a storage requiring merely 1.43 megabytes of storage. Whilst both such space requirements are easily satisfied by modern handheld devices, as the average smartphone possesses over 4GB of RAM (GSMArena, 2018), the run-time benefits do introduce a noteworthy improvement. Figure 14 showcases the speed-up provided by Bloom filters. The Bloom filter in questions uses the following parameters: [TODO: add params]. On a dataset of 100,000 revoked hashes, one can see that, as expected, the runtimes increase linearly. The x-axis varies the percentage of the candidates which are an actual member of the test data set. In other words, the percentage of actual matches increases in each subsequent measure. As expected, the verification utilising solely a Bloom filter is not impacted by this variation. Similarly, verification solely utilising Binary Search is also relatively unimpacted. The variation only utilising binary search on a possible match in the Bloom filter, is impacted the most. This variant only makes (expensive) I/O operations when the Bloom filter reports a possible match. As becomes apparent, the benefits from the Bloom filter decrease with the increase of the membership percentage. Hence, the speed-up is most prominent with lower membership percentage. In terms of attestation verification, a Bloom filter is thus most beneficent in case the vast majority of the encountered attestations are non-revoked and, thus, valid.

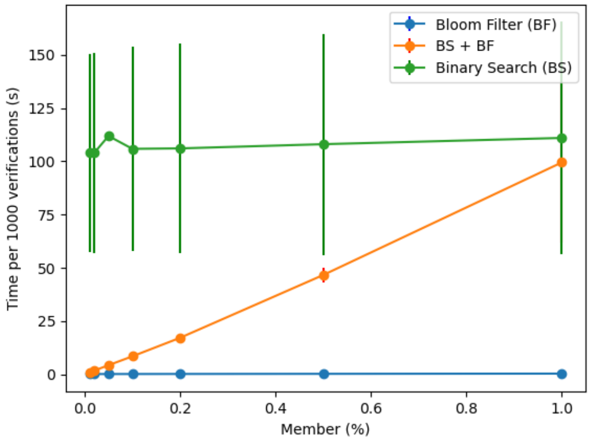


Fig. 14: Verification runtime per 1000 transactions (n=100,000

V. Related Works

A. Revocation

As a key contribution of our work lays in the field of revocation, we compare our works with the current state of the art in literature. We note that literature on revocation in Self-Sovereign Identity systems is not a widely discussed topic in academia, as such, the selected articles discuss revocation on a broader scale of digital identities. Table II displays the high level comparison of HRM compared to related revocation algorithms. The several comparison characteristics are quantified on a Low/Medium/High scale, where the actual performances are broad estimations based on the used technologies (e.g. blockchain). We note the selected related works either discuss more generic node revocation (Liau et al., 2005; Popescu et al., 2003) or are particularly tailored to Vehicular ad-hoc networks (VANETs) (Lasla et al., 2018; Haas et al., 2011). However, the general concept discussed in these works is the revocation of certificates, which could relatively trivially be transformed into the Self-Sovereign Identity domain as the expected loads of these systems can be shown to be compatible. For instance, Haas et al. (2011) assumes up to 25 million revocations, which is less than our assumed quantities.

As becomes apparent from Table I, HRM is expected to outperform previously proposed solutions on most characteristics. We note that HRM has a relatively higher storage requirement due to the storage of the raw revocations. However, as discussed previously, the storage can be narrowed down to a few megabytes through solely using a Bloom filter as opposed to additionally using cold storage. This, ofcourse, comes with the drawback of false positives and, as such, depends on the use-case of the client.

VI. Conclusion

We presented a Self-Sovereign Identity framework which can facilitate the digital identity needs of the European Union and can be deemed to be of *Industry Grade*. IG-SSI includes the possibility of attestation signing, presentation, verification and most notably, revocation. The scheme is shown to work fully distributed through the usage of IPv8 and allows for fully distributed revocation. Privacy is aided through the usage of zero-knowledge proofs and secure communication with specific peers can be forced through passphrases. A reference implementation for the semantic layer has been created, as well as a mobile client showcasing full usability on smartphones. Legally valid signatures can be achieved through IG-SSI, however, usability of the scheme has to be generated through global adoptation. For now, the results showcase that the characteristics of the system allow for a usable implementation.

References

Arias, D. (2021, Feb). *Adding salt to hashing: A better way to store passwords.* Auth0. Retrieved from `https://auth0.com/blog/adding-salt-to-hashing-a-better-way-to-store-passwords/`

Bloom, B. H. (1970). Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM*, *13*(7), 422–426.

Boneh, D., Goh, E. J., & Nissim, K. (2005). Evaluating 2-DNF formulas on ciphertexts. In *Lecture notes in computer science* (Vol. 3378, pp. 325–341). Springer Verlag. Retrieved from `https://link-springer-com.tudelft.idm.oclc.org/chapter/10.1007/978-3-540-30576-7_18` doi: 10.1007/978-3-540-30576-7{_}18

Boudot, F. (2000). Efficient Proofs that a Committed Number Lies in an Interval. In B. Preneel (Ed.), *Advances in cryptology — eurocrypt 2000* (pp. 431–444). Berlin, Heidelberg: Springer Berlin Heidelberg.

Buterin, V. (2013). A next generation smart contract & decentralized application platform. *Ethereum Foundation.*

by Design Foundation, P. (n.d.). *Privacy by design foundation.* Author. Retrieved from `https://privacybydesign.foundation/irma-explanation/`

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

European Commission. (2021, Jun). *Eu digital covid certificate.* European Commission. Retrieved from `https://ec.europa.eu/info/live-work-travel-eu/coronavirus-response/safe-covid-19-vaccines-europeans/eu-digital-covid-certificate_en`

GSMArena. (2018, 4). *Counterclockwise: RAM capacity through the years* . Retrieved from `https://www.gsmarena.com/counterclockwise_ram_capacity_through_the_years-news-30756.php`

Haas, J. J., Hu, Y. C., & Laberteaux, K. P. (2011, 3). Efficient certificate revocation list organization and distribution. *IEEE Journal on Selected Areas in Communications*, *29*(3), 595–604. doi: 10.1109/JSAC.2011.110309

Halkes, G., & Pouwelse, J. (2011). UDP NAT and firewall puncturing in the wild. In *Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics)* (Vol. 6641 LNCS, pp. 1–12). Springer, Berlin, Heidelberg. Retrieved from `https://link-springer-com.tudelft.idm.oclc.org/chapter/10.1007/978-3-642-20798-3_1` doi: 10.1007/978-3-642-20798-3{_}1

HM Passport Office, H., Border Force, & The Rt Hon Caroline Nokes MP. (2018, Jun). *Report your lost or stolen passport.* GOV.UK. Retrieved from `https://www.gov.uk/government/news/report-your-lost-or-stolen-passport`

Khovratovich, D., & Law, J. (2017). *Sovrin: digital identities in the blockchain era* (Tech. Rep.). Retrieved from `http://www.credentica.com/the`

Koens, T., Ramaekers, C., & Van Wijk, C. (2018). *Efficient Zero-Knowledge Range Proofs in Ethereum* (Tech. Rep.). ING. Retrieved from `https://www.ingwb.com/media/2122048/zero-knowledge-range-proof-whitepaper.pdf`

Kwiatkowska, M., Norman, G., & Parker, D. (2008). *Analysis of a Gossip Protocol in PRISM* (Tech. Rep.). Retrieved from `http://www.prismmodelchecker.org/casestudies/gossip.php`

Laberteaux, K. P., Haas, J. J., & Hu, Y.-C. (2008). Security certificate revocation list distribution for vanet. In *Proceedings of the fifth acm international workshop on vehicular inter-networking* (pp. 88–89).

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

Liau, C. Y., Bressan, S., & Tan, K.-L. (2005). Efficient Certificate Revocation : A P2P Approach. *HICSS'05.*

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

Nakamoto, S. (2009). *Bitcoin: A Peer-to-Peer Electronic Cash System* (Tech. Rep.). Retrieved from `www.bitcoin.org`

Peng, K., & Bao, F. (2010). An efficient range proof scheme. In *Proceedings - socialcom 2010: 2nd ieee international conference on social computing, passat 2010: 2nd ieee international conference on privacy, security, risk and trust* (pp. 826–833). doi: 10.1109/SocialCom.2010.125

Popescu, B. C., Crispo, B., & Tanenbaum, A. S. (2003). A certificate revocation scheme for a large-scale highly replicated distributed system. In *Proceedings - ieee symposium on computers and communications* (pp. 225–231). doi: 10.1109/ISCC.2003.1214126

Raya, M., Jungels, D., Papadimitratos, P., Aad, I., & Hubaux, J.-P. (2006). *Certificate Revocation in Vehicular Networks* (Tech. Rep.). Retrieved from https://www.researchgate.net/publication/37433732

Raya, M., Papadimitratos, P., Aad, I., Jungels, D., & Hubaux, J.-P. (2007). Eviction of Misbehaving and Faulty Nodes in Vehicular Networks. *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS*, *25*(8). Retrieved from http://www.sevecom.org doi: 10.1109/JSAC.2007.0710xx

Siftery. (2017, Jan). *Top social login tools compared.* Medium. Retrieved from https://medium.com/@siftery/top-social-login-tools-compared-b350eae26118

Sporny, M., Longley, D., & Chadwick, D. (2019, Nov). *Verifiable credentials data model 1.0.* W3C. Retrieved from https://www.w3.org/TR/vc-data-model/

Stigler, G. J. (1964). A theory of oligopoly. *Journal of Political Economy*, *72*(1), 44–61. Retrieved from http://www.jstor.org/stable/1828791

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).

Von der Leyen, U. (2020, 9 16). *State of the union address by president von der leyen at the european parliament plenary.* Retrieved from https://ec.europa.eu/commission/presscorner/detail/en/SPEECH_20_1655

Zeilemaker, N., Schoon, B., & Pouwelse, J. (2013). *Dispersy bundle synchronization* (Tech. Rep. No. PDS-2013-002). TU Delft. Retrieved from http://www.pds.ewi.tudelft.nl/fileadmin/pds/reports/2013/PDS-2013-002.pdf

Zhou, T., Li, X., & Zhao, H. (2019). EverSSDI: Blockchain-based framework for verification, authorisation and recovery of self-sovereign identity using smart contracts. *International Journal of Computer Applications in Technology*, *60*(3), 281–295. doi: 10.1504/IJCAT.2019.100300

TABLE II: Revocation comparison with related works

	HRM (this work)	**Lasla et al. (2018)**[1]	**Popescu et al. (2003)**	**Liau et al. (2005)**	**Haas et al. (2011)**[2]	**Laberteaux et al. (2008)**
Offline availability	✓	✓	✗	✓	✓	✓
No Authority interactivity	✓	✓	✗	✗	✓	✓
Storage requirement	**Med**	High	Low	High	Med	Med
Processing requirement	**Low**	High	High	Low	Low	Low
Revocation latency	**Low**	High	Low	Low	Med	Med
Verification latency	**Low**	Low	Med	High	Low	Low
No SPOF	✓	✓	✗	✓	✓	✓
No False positives	✓	✓	✓	✓	✗	✓
No False negatives	✓	✓	✗	✓	✓	✓
SSI Compatible	✓	✗	✗	✗	✗	✗

[1] As no specification on the type of blockchain was given, we assume the usage of the Bitcoin blockchain as per their test results.

[2] As the propagation of revocations is possibly dependent on both the distribution through Vehicle-to-Vehicle communication and Road-Side Units, we assume a revocation latency higher than direct client-to-client communication and lower than that of a (PoW) blockchain.

II

Supplementary Material

The second part of this thesis comprises supplementary material serving as an accompaniment to the thesis article of Part I. This material is composed of four appendices: extended background information, implementation details, results, and conclusion. In background information, additional insights into the concept of SSI and digital identities are given. Foremost, this chapter aims to create an academic definition of SSI.

A

Background Information

As Self-Sovereign Identity is a relatively new scientific field with its origin outside of academia, this section aims to provide the reader with an understanding of the concept. Firstly, the terms identity and digital identity are discussed as well as the history of Digital Identity Management Systems. Next, inherited flaws of the current eco-system are discussed. Finally, the different definitions of SSI are discussed and our theoretical framework encapsulating them is introduced.

A.1. Identity

Identity has a broad spectrum of definitions. The terminology itself stems from the Latin word for sameness, namely identitās (Merriam Webster, n.d.). Philosophy draws the distinction between qualitative and numerical identity (Noonan & Curtis, 2018). Qualitatively, identity is defined as entities sharing certain characteristics. Whilst numerically, we speak of total qualitative identity, thus requiring a set of characteristics which an entity only shares with itself. These characteristics are referred to as attributes (Camp, 2004). The notion of the numerical identity of a person through time is referred to as the personal identity (Olson, 2021). The foundations of this law can be traced back to Aristotle's Law of Identity, broadly stating that everything is equal to itself (Aristotle, 1925).

The requirements for the technical sense of identity are most fulfilled by the definition of the numerical variant. As it can be said that the goal of digital identity is to uniquely identify entities. Hence, personal identity may prove to fall short in such specification, as digital identity does not solely consider persons. Namely, ISO, 2019 defines identity as "any set of attributes that describe a particular entity". We can, thus, state that identity is the set of characteristics uniquely describing an entity. Hence, we make no distinction between human identity and the identity of software-based entities (e.g. Artificial Intelligence or bots)

When such a characterisation is transformed to the digital domain, we speak of digital identity. The goals of digital identity are identification and authentication (Bertino, 2006). Where identification can be seen as the authorisation of one's identity (Camp, 2004) allowing the unique identification of a user in a system (IBM, 2021). Authentication is the action of proving one's identity. This can be achieved by three means:

- Something you know (e.g. a password).
- Something you have (e.g. a smartcard or key).
- Something you are (e.g. biometrics: fingerprint, face, etc.).

Often, measures are combined, referred to as multi-factor authentication.

A.2. Digital Identity

The Internet was not created with an identity layer. Even the conceptual OSI-model (Zimmermann, 1980) does not contain a layer specifically designed for identity. As a consequence, there is no digital identity. The current digital eco-system comprises one's digital presence through fragments of pseudo-identities. These pseudo-identities ultimately belong to a single entity and, thus, all attempt to be a digital identity. Of course, one is able to be identified digitally through these shards. However, these pseudo-identities lack the knowledge to fully uniquely identify an entity. We refer to this phenomena as the Sharding of Identity. Each of these pseudo-identities often attempt to authenticate the same data. For instance, name, age, and a means of communication (e.g. e-mail). As such, all these pseudo-identities can be labelled as being derivatives of one's actual identity: the true digital identity. One that is uniformly true and does not require indefinite copies for each new encounter. The relationship between these groups are visible in Figure A.1. As is visible, one's digital identity is a subset of one's physical identity, indicating that the digital identity is invariably linked to the entities physical identity, and the group of digital identity shards is a subset of what one's digital identity may possibly be. We note overlap between identity shards, which is caused by a non-empty union of the attributes comprised by said shards. For instance, the vast majority of services require a registration per name. As such, most digital identity shards will have at least an overlap on this attribute. As may become apparent from this description, these digital pseudo-identities fall under qualitative identity as most of them share attributes with other shards. This follows naturally from the fact that each of them attempt to identify the same entity.

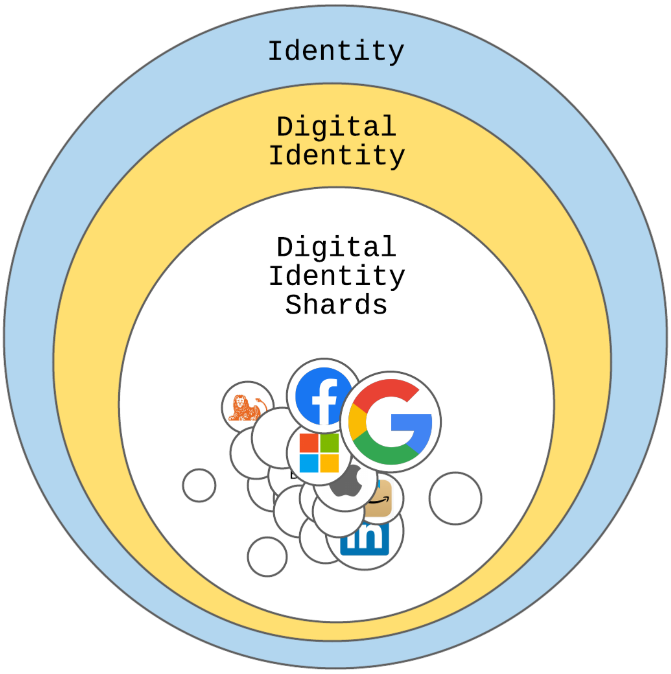


Figure A.1: Identity Groups

Cameron, 2005 describes some aspects of this phenomena. Cameron refers to the Internet as “a patchwork of identity one-offs". With this statement, they refer to the same phenomena that resulted in the Sharding of Identity. Namely, each Internet service providing or requiring and managing its own (unique) identity system. Stokkink and Pouwelse, 2018 refers to these shards as “digital identity schizophrenia”.

A.3. The Evolutions of DIMS

Allen, 2016 describes four phases of digital identity. Although the chronological ordering of the phases is correct, we argue that the term phase is not correct for these specifications as phases indicate non-concurrent existence. For instance, the eight phases of the Moon do not exist simultaneously. Therefore, we propose the usage of the term evolution. As evolution indicates gradual development, whilst allowing simultaneous existence with prior iterations. Note that evolution does not necessarily indicate improvement (Hall et al., 2008), which is also not insinuated by the term phases. Hence, the following four evolutions of digital identities exist:

Evolution One: Centralised Identity

With the onset of the Internet, centralised authorities such as IANA[1] and ICANN[2] became the issuers and authenticators of digital identities. For instance, the IANA determined the validity of IP addresses (IANA, n.d.), whilst the IANA managed the registration of domain names (ICANN, 2017). 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 online services implementing their own digital identity management systems, which for the user often led to username and password combinations. All of the aforementioned organisations present in the Internet ecosystem are inherently centralised authorities, with capabilities of revoking these identities. This comes with the consequence of users not owning any of their digital identities, as they are all either assigned to them or are managed by others. For instance, the registration of a domain name is performed on a yearly-bases (ICANN, 2017), allowing one to never fully own a domain name.

Evolution Two: Federated Identity

The second generation attempted to overcome the hierarchies, by imagining a federated identity. An example of this is Microsoft's Passport initiative (PressPass, 1999), allowing identities across different domains. However, this initiative soon proved to be far from optimal, as it is comprised of a single authority. This was improved upon by allowing each site to remain an authority (Allen, 2016). However, users were still not provided with the means of controlling what happened with their data. Hence, there was a need for a new evolution catering to the user aspect of digital identities, as opposed to the identity management aspect.

Evolution Three: User-Centric Identity

Currently, identity management systems are in the third generation, the “User Centric Identity Management", originally described by Jøsang and Pope, 2005 . This generation attempts to put the user at the centre of their identity. Open-sourced examples of these include OpenID[3], OAuth[4] and FIDO[5]. These systems focus on user centricity through consent, and interoperability, allowing users to select their own provider. Unfortunately, these efforts have resulted in the still register being the owner of the identity, instead of the user. However, the main drawback to the current phase is the introduction of initiatives such as Facebook ConnectMorin, 2008 (contemporary known as Facebook Login[6]) or Google Identity[7]. Whilst these initiatives do allow selective sharing of identity information and regard user consent, they still store identities in a centralised fashion and are managed by a single authority, namely a commercial party. The global adoption of these digital identity providers, has led to what we refer to as, the oligopoly of digital identities.

Allen refers to ano source (?)

The oligopoly poses additional threats to users. The main issues regarding this oligopoly are lack of control, privacy, and information asymmetries. More prominently, Big Tech now has the ability to potentially revoke ones digital identity without warning, resulting in a loss of access to possibly countless of services. Privacy is at peril as Big Tech is enabled to gain information on their users through other services. This privacy concern can lead to market mechanisms such as information asymmetries, due to these extra opportunities for data farming. 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"[8] 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. These concerns portray a need for a different approach to identification, breaking the oligopoly and creating the ability to generate trust over the Internet.

[1] For IANA, see: https://www.iana.org/

[2] For ICANN, see: https://www.icann.org/

[3] For OpenID, see https://openid.net/connect/

[4] For OAuth, see https://oauth.net/

[5] For FIDO, see https://fidoalliance.org/

[6] For Facebook Login, see: https://developers.facebook.com/docs/facebook-login/

[7] For Google Identity, see: https://developers.google.com/identity

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

A.4. Shortcomings in the Current Ecosystem

The current ecosystem of digital identities suffers from several drawbacks and limitations, both from the perspective of identity providers as of that of the users.

A.4.1. Problems for Identity Providers

For identity providers, identification measures can prove to be a double-edged sword: whilst it allows them to manage their users' digital identities, allowing them to gather user statistics and information in order to improve their services, it can also prove to be a burden. Firstly identity providers must adhere to specific data compliance legislation such as the GDRP (The European Parliament and Council, 2016) or the PCI DSS (Council, 2004). Additionally, often companies strive for internation standards such as ISO/IEC 27001 (ISO, 2013). The leakage of Personal Identifiable Information (PII) cannot only lead to liability in accordance to said legislation (e.g., the GDPR has the possibility to fine companies in the millions), but can also have side effects for the users. For instance, in case passwords are compromised, other services utilised by the user may be at peril or the leaked PII can be used for spear-phishing attacks. Moreover, such losses can have tremendous impact on the image of an organisation. Breaches such as the Cambridge Analytica Scandal Rosenberg, 2018, portray the impact.

A.4.2. Problems for Users

On the other end, users suffer from these consequences and more: firstly, users must keep track of all their fragmented digital identities, often requiring to manage a multitude of digital identities. A report published by LastPass in 2019, shows that on average employees of small businesses manage 85 passwords. With the statistic that the use of brute-forced or stolen credentials are responsible for over 80% of the vulnerabilities utilised in breaches (Verizon, 2020), credentials continue to be a weakness in online identification measures. Secondly, users' information is stored in numerous amounts of locations, significantly increasing the chances of their PII to be stolen, as this increases the attack surface. For instance, Thales, 2020 reported that in 2020, 49% of US companies reported a digital breach of some degree.

Furthermore, the oligopoly poses additional threats to users. The main issues regarding this oligopoly are (I) a disproportional balance of power, (II) privacy issues, and (III) information asymmetries.

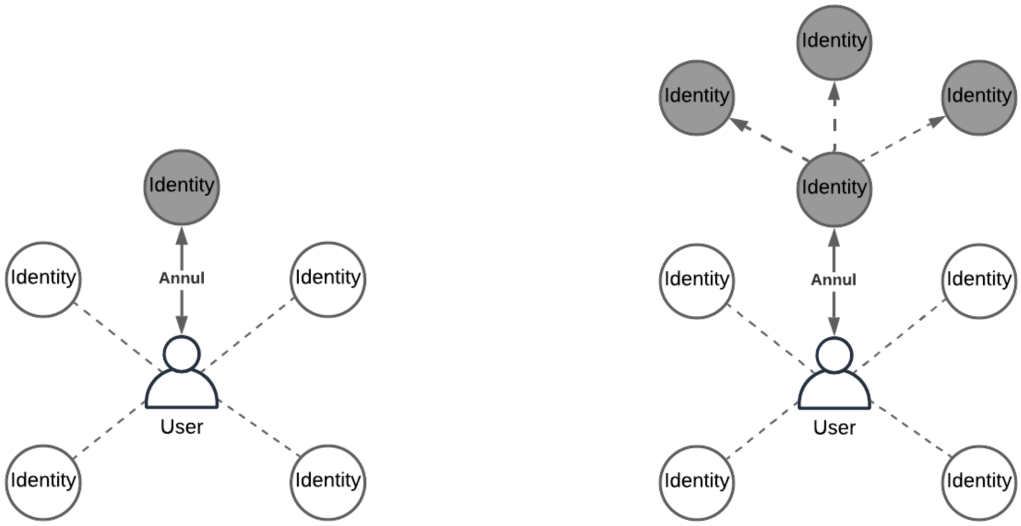
The Balance of Power

The disproportional balance of power is caused by the connection with other services. In a central identity, i.e. one in which the service provider is also the identity provider, the user and the service provider hold relatively the same amount of power. More specifically, the user has the ability to stop their usage of the service and, thus, losing a single digital identity shard. Similarly, the service provider has the ability to refuse service to the user, revoking in turn a single digital identity shard and, thus, revoking access to a single web service. This generates a balance of power within their relationship, as both of their abilities to annul the digital identity lead to a single loss. Hence making the balance one-to-one. This has been visualised in Figure A.2a, portraying a one-to-one annulment relationship. Of course, this lays more delicately, as often the service provider generates value for the user making the user reliant on them to some degree, hence, shifting this balance in the favour of the provider.

When the identity provider manages a federated identity system or a user-centric variant, this balance shifts greatly. As now, a user desiring to annul his digital identity with such a provider, will cascadingly annul his access to any connected service. Hence, they are often not able to discontinue any arrangement with them without affecting their arrangement with other service providers. On the other hand, as the identity provider has the ability to revoke ones digital identity, users may face loss of access to any connected services. For instance, in case a user is deemed to have breached a term of use. This has been visualised in Figure C.2b, which portrays the imbalance of annulment power.

Privacy Issues

IBM, 2019 shows that 84% of the people believe that they have lost all control over regarding the usage of their data by companies. This aids in portraying the privacy issues experienced by users. The digital identities managed by Big Tech can further impact privacy, as they enable the gathering of more information on their users through other services. Any connected service has the potential to serve as a funnel for additional data



(a) Centralised Identity

(b) Federated/User-centric Identity

Figure A.2: Balance of Power

on user information. The 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.

Information Asymmetries

The privacy concerns can lead to market mechanisms such as information asymmetries, due to the extra opportunities for data farming. As the identity providers posses large amounts of information on 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 interest. These concerns portray a need for a different approach to identification, breaking the oligopoly and creating the ability to generate trust over the Internet.

A.5. Theoretical SSI Models

As no consensus on a formal definition of Self-Sovereign Identity has been reached, the properties of SSI are loosely defined. There are, however, 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.

A.5.1. The Laws of Identity

As mentioned previously, Cameron describes the seven laws of identity, which DIMS are to adhere to (Cameron, 2005). Whilst not directly calling for sovereignty over digital identity, the majority of principles described by Cameron can be identified in contemporary notions of SSI, hence, we can make the case that SSI was created in 2005, at least the foundations for it. The following laws are proposed by Cameron, 2005:

1. **User control and consent**: DIMS' must only reveal personal information given prior consent by the user. We shall refer to this personal information as PII. 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 ofPII, 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 DIMS' must support "omni-directional" identifiers, 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 DIMS' must support multiple identity technologies ran by multiple identity providers. This law enables the incorporate this somewhere, disallowing vendor lock-in and encouraging the use of open-standards.

gically agnostic

6. **Human integration**: universal DIMS' 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 DIMS' 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.

Whilst not coining a specific term for such a system, we do identify key aspect relevant to SSI which were later—in an adapted form—reiterated in the conceptualisation by Allen, 2016.

A.5.2. The Path to Self-Sovereign Identity

Allen, 2016 is undeniably the most commonly referred to literature with respect to SSI. In their work, the following set of principles are posed:

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, exist until destroyed by the user.

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

7. **Interoperability**: identities must be as widely usable as possible. This ensures that digital identities can be globally deployed. This properties is aided by the Transparency principle, as open-standards allow for more seamless integration of other systems.

8. **Consent**: users must agree to the use of their digital identity. This principle strengthens the Control principle, as the sharing of attributes may only occur with the consent of the user. It is noted that this should 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 focuses on privacy and prevents misuse of data.

10. **Protection**: the rights of users must be protected. The rights of users must take precedence over the identity network itself. This can be achieved through 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 to the human, which is the most important entity in the system. Furthermore, human control is key to the design. We note a large overlap with the work of Cameron, 2005. The laws “User control and consent"; “Minimal disclosure for a constraint use"; “Pluralism of operators and technologies"; “Human integration"; and “Consistent experience across context" can be directly identified from the ten principles from Allen, 2016. 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 builds upon these ten principles by subdividing them 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.

A.5.3. The Pyramid of Sovereignty

The previous sections portray a crisis in terms of both definition and the naming of the principle that is referred to as Self-Sovereign Identity. We believe that this is mostly caused by the unacademic origin of SSI. As such, we propose a new set of principles based on the commonly cited works of Allen, 2016; Cameron, 2005, however, also taking into account the literature that sketched the beliefs of SSI Loffreto, 2012, 2016. We propose the pyramid of sovereignty as presented in Figure A.3.

The main pyramid consists of ten principles, having overlap with Allen, 2016 and Cameron, 2005. The corner stones of the framework are existence and control. Existence requires an SSI to be linked to an entity. Where most literature requires a link with a human identity Loffreto, 2016; Speelman, 2020, we state that an SSI must be linked to an entity in order to exist. We argue that this is a necessity for the prosperity and longliveness of SSI as a whole. Especially for the ongoing fourth industrial revolution (Moore, 2019), in which SSI can prove to fulfil a prominent role with the communication of IoT Sovrin, 2019. Hence, other entities such a IoT devices, bots or artificial intelligence, which facilitate any form of communication with humans or support systems, can prove to gain valuable characteristics through SSI. Control enables the user-centric nature of SSI, allowing complete access, consent, and usage of the data stored by an SSI for the user. Allen, 2016 splits this up in control, access and consenst. However, we argue that control implies requirement of consent, as in full control no action is to be performed without knowledge of the one in control. Similarly, we argue that control implies access. Furthermore, control must also imply a free choice of storage. Hence, we deem the term control sufficient for enabling the user-centric nature. Furthermore, the bottom layer of the pyramid is reinforced by verifiability: a property not explicitly mentioned as a required in most literature, apart from Stokkink et al., 2020. However, we deem verifiability to be one of the main foundations of SSI. As without verifiability, information holds no value.

The second layer consists of transparency and interoperability. Where transparency strives for the usage of open standards and implementations, of which the very least the details of used algorithms and protocol are openly defined. This aids in making SSI a common good and ensuring that the principles are adhered to. interoperability ensures that a user is not locked in a specific implementation of SSI, allowing the communication with other services and possibly other systems. This aids in both ensuring user’s rights as well as the adoption of SSI through easy usage with existing solutions.

The third layer is comprised of minimalisation: this principle ensures that no more information is shared than is required. This also entails that no information is shared with parties that do no explicitly require it. This falls in line with the comparable law posed by Cameron, 2005.

The final layer consists of privacy. The combination of all previous layers allows one to achieve a certain degree of privacy. Of course, no full privacy is achievable when sharing personal data. However, the SSI system must attempt to guarantee a certain level of privacy. Which is especially reinforced by the control, transparency and minimalisation principles.

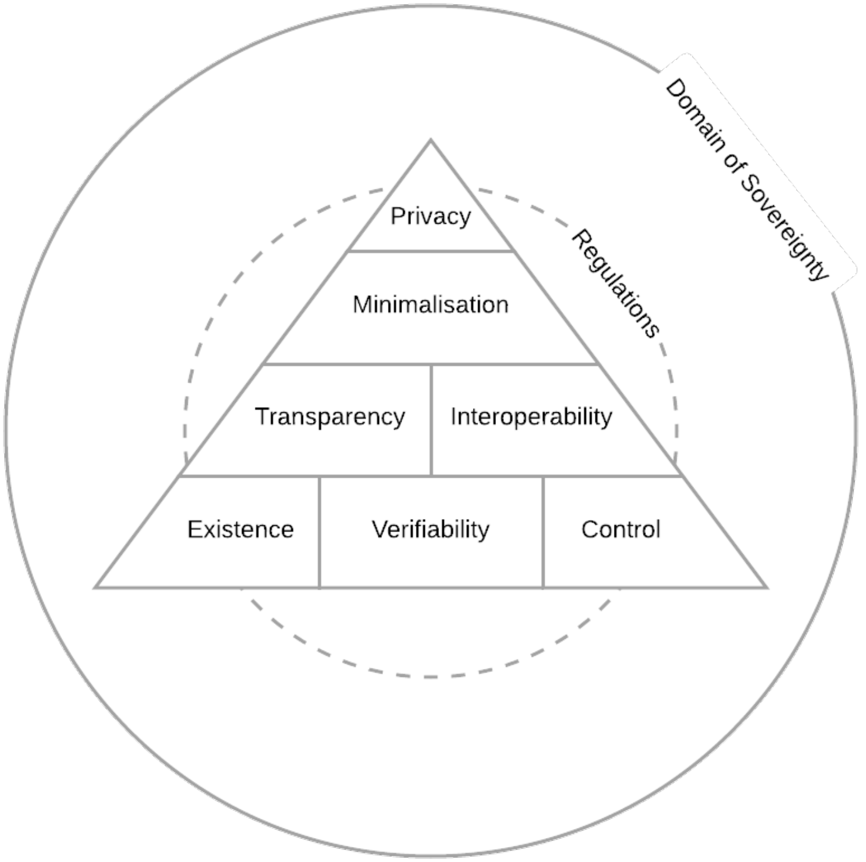


Figure A.3: The Pyramid of Sovereignty

Finally, the Pyramid is contained by two shells. The inner shell represent regulations imposed upon the system by for instance governments. In case legally valid credentials are introduced, legislation comes into play. This will most likely counteract, or at least deteriorate, the strength of some of the principles. This is visualised by the intersecting nature of the inner shell. The outer shell represents the Domain of Sovereignty, in which the further the pyramid nears the bounds of the domain, the higher the degree of sovereignty. As mentioned previously, Loffreto, 2012, 2016 describes a more anarchic nature of SSI than most other literature envisions SSI to be. The outer bounds of this domain represent this level of sovereignty. As is visible, the proposed framework is more restricted in its levels of sovereignty than the notions discussed by Loffreto.

A.6. Existing Solutions

A.6.1. History of Self-Soveregn Identity

There is no clear onset of Self-Sovereign Identity. Allen, 2016; Preukschat and Reed, 2021 refer to the work of “What is ‘Sovereign Source Authority’?" (Loffreto, 2012) to be the onset of SSI. Allen misatributes this work to Moxie Marlinspike, the co-founder of Signal[9]. However, presumably this was performed on purpose (Sheldrake, 2016). In their work, Loffreto describes the concept of Sovereign Source Authority (SSA). With SSA, Loffreto calls for an overhaul of the current national administrative identities. They refer to the current systems as lacking the ability of providing a real identity, as current identities can be seen as a registration process for participation in society. SSA can be seen as a need for what Loffreto refers to as Human Identity. This falls in line with the Identity Groups as discussed in section A.2.

Loffreto’s main argument for an alternative identity system is comprised of societal participation being a choice, hence one must be able to have a valid identity without participating in society. Loffreto states

[9] For Signal, see: https://signal.org/

Table A.1: The principles by Loffreto (2016)

Principle	Description
Human Life	An SSI originates from an individual human life.
Human Identity	The human identity is the source authority of an SSI.
Attestations	An SSI has no personal control or authority until it is attested to by others.
Unpragmatic	SSI is not to be pragmatically defined as it is a function of time and place.

that "Within any Society, Individuals have an established Right to an 'identity' ". The term Sovereign Source Authority itself did not gain much traction. However, it did lead to the coining of the term Self-Sovereign Identity. Whilst no key literature has been identified for coining the term itself, it can be said that SSI has gained traction due to Christopher Allen. Allen has often been erroneously credited for the invention of the term Self-Sovereign Identity, however, has explicitly credited Loffreto. Four years later, in 2016, Loffreto made another blog post on explicitly SSI. Loffreto, 2016 describes four properties of SSI, which have been summarised in Table A.1. As becomes apparent from this description, Loffreto does not consider SSI to be a digital technology, but more a concretization of the human life, capable of authenticating the human identity.

Later in the same year, Allen released their blog post "The Path to Self-Sovereign Identity" Allen, 2016, which, undeniably has been a major influence on the field, with all major publications referencing said work, e.g. Baars, 2016; Ferdous et al., 2019; Mühle et al., 2018; Stokkink and Pouwelse, 2018; Tobin and Reed, 2016. In their work, Allen, 2016, describes ten principles which SSI is to adhere to. However, an often uncredited piece of literature in SSI is that of Cameron, 2005's "Laws of Identity", where Laws is used in the scientific sense. In this work, published more than a decade prior to any literature directly referencing SSI, Cameron calls for a need for a "unifying identity metasystem" Cameron, 2005. Furthermore, the concepts of digital subjects and claims are introduced, making way for claim-based identities. This work describes a large number of fundamentals of SSI and is often disregarded in literature on SSI, whilst being a highly influencing article in DIMS in general, with laws being implemented in systems such as OpenID 2.0 (Recordon & Reed, 2006). These laws explain the shortcomings and successes of digital identity systems and, as such, are applicable to SSI. The works of Allen, 2016; Cameron, 2005; Loffreto, 2016 are described more thoroughly in section A.5

A.6.2. Critique of the Term

Sovereignty is defined as "supreme authority within a territory" (Philpott, 2020). In terms of Self-Sovereign Identity, this would translate to supreme authority over one's identity. This term is prone to misinterpretation. As supreme authority insinuates that one has the full power of some territory. However, the extent to which this power reaches is open for interpretation. For instance, Good ID, 2021 defines Self-Sovereignty as "a feature of an ID or identity system, whereby, individual users maintain control over when, to whom, and how they assert their identity". There exists a discrepancy between this definition and the definition created using Philpott, 2020. We identify the same discrepancy in literature. For instance, the works set out by Loffreto, 2012, 2016 portray a philosophical nature of SSI, not necessarily indicating the usage of DIMS. DIMS are merely an implementation which allows a realisation of SSI. Furthermore, proposed solutions such as Ferdous et al., 2019; Khovratovich and Law, 2017; Lundkvist et al., n.d.; Zhou et al., 2019 do not necessarily adhere to this description. However, the case can be made that Stokkink et al., 2020; Stokkink and Pouwelse, 2018 as well as, transitively, IG-SSI would allow for the human identity as origin of source authority, as described by Loffreto, 2016. It can be noted that SSI and DIMS are undeniably intertwined, having led to misinterpretations of the term itself. Concerns for the usage of the term have been raised (Cameron, 2018; Ruff, 2018). Common misconceptualisations due to the term itself, are the following Ruff, 2018:

- **Self-sovereign means self-attested**
  The term sovereignty implies total domination and, as such, could lead to self-attestation. However, even in the descriptions proposed by Loffreto, 2016, claims require attestations. We do believe that verifiable claims allow for self-attestations, as in certain instances verifiability through others is simply not required. However, it is not the case that any self-attested nature is a given.

- **SSI attempts to reduce government's power over an identity owner**
  This claim we deem invalid due to a multitude of reasons. Firstly, SSI, as is generally true for any form

of technology, is adiaphorous; SSI is not an entity that can act, hence it is inherently neutral. The realisation and usage of SSI could impact a government's power in the identity domain due to shifts in ownership. Loffreto, 2012, 2016 do propose SSI as an alternative to the centralised governmental identities, as they deem the centralised registration unnatural. Secondly, the case can be made that the traditional governmental identity can evolve into SSI. With active plans from the European Commission to introduce a European Digital Identity, wide-spread SSI may even be introduced by government Comission, 2021. Moreover, SSI can prove to not delegate any power from governments as they can simply become an attestor for a digital identity credential, making them intrinsically an authority. As government can be considered a commonly accepted authority, the network will most probably acknowledge—and even require—the government for verification of a digital identity. SSI can even prove to aid governments by reducing the need for maintaining the physical identification documentations.

- **SSI gives absolute control over identity**
  This misconceptualisation is most likely caused due to the ambiguous nature of the term Self-Sovereignty. As we established that self-sovereignty does not lead to self-attested, the dependency on attestation directly deteriorates the level of control one has over claims. Verifiable claims require authorities to attest to a certain piece of information. The lack of authoritarian party, which is deemed trusted amongst the network, that attests to a claim, making it verifiable, leads to a weak claim. Hence, whilst one does have the power to self-attest and, furthermore, to fully control the actions regarding his data in terms of verifiable claims, one's self-sovereign identity will still be dependent on others. For instance, it is a possibility that a digital identity will only become valid in case it is attested to by a governmental institution, as otherwise there is no neutral party in which one can built trust in the validity of the information of the claim. Intrinsically, there is no true sovereignty over what attributes one has, but merely, sovereignty over what happens with said attributes.

The above critiques and misconceptualisations sketch the ambiguous nature of the sovereignty side of SSI. It leads to a need for a more defining term. Ruff, 2018 proposes the use of decentralised identity as an intermediary term. However, the major shortcoming of this term is that it does not convey the level of control that a user has. As, in order to be classified as decentralised, a system must simply consist of multiple parts which collaborate in order to achieve some goal. Hence, the selection of decentralised identity is not strict enough in order to explicitly convey the users' rights. Therefore, we propose the usage of the term Self-Governed Identity (SGI) in order to specify and distinguish what literature most commonly refers to as SSI from the more anarchic SSI discussed by Loffreto, 2012, 2016. To govern can be defined as "conduct the policy, actions, and affairs of (a state, organization, or people) with authority" (The Oxford Dictionary, n.d.). When placing this definition in the context of identity, one would be able to conduct the policy, actions, and affairs of one's identity. This constraint the power of the principles behind SSI, as self-governed does not imply total dominion over one's identity as sovereignty does. As a consequence self-governed implies that one has full control of one's identity, whilst not necessarily defining what the identity is. This flows naturally from the instances in which identity is to be assigned to one. As any society decides that a government has the power to delegate identities, hence, SSI will most likely have to adhere to this structure in order to gain any form of legal validity. However, this does not mean that SSI itself must force this behaviour, as self-attestations have valid use-cases. Even more, an identity may be formed through self-attestation. However, this most likely does not lead to any legally valid identity as such an identity is only valid to the extend that it is accept by any parties with whom one will communicate and require identification to. A government, hence, provides a (relatively) neutral party in which one can generate trust in order to only attest to valid identities, allowing verification of such identities. Hence, the term Self-Governed Identity can prove to encapsulate this almost inherently unavoidable unbalance of power. SSI can, hence, be seen as a digital alternative as opposed to a digital revolution, as it is unavoidable that certain existing and, most probable, required power balances must be transformed to the digital domain in order to safeguard the identity. However, we believe that the most important nature of SSI and, subsequently, the more lenient proposition of SGI is to place data back in the hands of users and providing the digital domain with legally valid identities and verifiable information without the need for a central authority. However, in the remainder of this document the term SSI is used as opposed to SGI in order to be in line with the majority of the literature.

A.6.3. Sovrin

The Sovrin Foundation[10], 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[11] standard from W3C (Reed et al., 2016). In order to aid privacy, each relation uses new public and private keys

find citation

A.6.4. Serto (uPort)

. Serto[12], formally known as uPort, is an SSI solution built on Ethereum (Braendgaard, 2017; Lundkvist et al., 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 (**WhatMedium**). The underlying structures are built on the JSON format.

A.6.5. 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)[13]. 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. as visible in Table A.2:

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

A.7. Related Works

Incorporate this m
seamlessly

[10] For Sovrin, see https://sovrin.org/

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

[12] For Serto, see https://www.serto.id/

[13] For World Wide Web Consortium, see http://w3.org/

Table A.2: The properties defined by DID

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

A.7.1. 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[14]. 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.

A.7.2. Der et al., 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.

[14] https://identity.foundation/

A.7.3. 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 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.

A.7.4. 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.

A.7.5. Ferdous et al., 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).

A.7.6. Zhou et al., 2019

Zhou et al., 2019 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.

A.7.7. Belchior et al., 2020

Belchior et al., 2020 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.

A.8. Ethical Implications

The previously discussed properties and use-cases portray an alternative to the current ecosystem of DIMS with potentially far-reaching implications. As such, we deem it necessary to discuss the ethical implications of such a system. For this, we portray two scenarios regarding the implications of the deployment and adoption of such a system.

A.8.1. The End of Privacy

A major case for SSI is the introduction of legally valid digital signatures, a concept which is not yet properly defined by legislation. In case governments back such a system, the possibility for legally bound digital identities opens up. It can be said that the current ecosystem of managing digital identities can be quite cumbersome for online services, as such they may opt to require such a system. A major benefit for this is the possibility of eliminating bots and spam accounts. Without a digital identity attested to by a government, a platform may choose to deny service to such users. After all, in case SSI is adopted by the government there is no reason that a human has no access to such an identity. In case each service requires authentication through SSI, we can speak of an end to digital privacy. As each server now is able to uniquely identify you to a legal extend. Furthermore, this makes tracking across context almost perfectly viable. This opens up the possibility for more data farming and targeted advertisements. Far less benign would be further personalised spear phishing attempts.

As such, a certain degree of anonymisation should be implemented in order to prevent such a scenario. For instance, services should not require your full legal name apart from when they are obliged by legislation to gather such information (e.g. an e-commerce platform or a financial service provider). As such, the presentation of attestations in zero-knowledge can prove to provide such a degree of anonimisation.

A.8.2. Yet-Another-Authentication-System

On the other end of the spectrum, SSI can prove to become not more than a new authentication system, being used along side the current measures. The user-centric and federated identities are far from fully accepted by every service. As such, the critical question must raised whether the next evolution will be the dominant system and, thus, be fully supported by the majority of online services and even physically. We argue that this is dependent on the backing of any governmental agencies. Whilst not a necessity, governmental adoption can prove to be the major influence on the widespread adoption of the technology. In this sense, we argue that governmental adoption can lead to legal validity of attestations and, therefore, passport grade digital authentication. However, we do not that such attestations can also be achieved through a company serving as an authority and performing verification of one's identity through e.g. government-issued identification documentation. However, this would then most probably lack any legal foundation. In such a scenario we deem it more likely that SSI would be yet-another-authentication-system.

B
Design

This chapter describes the design of the Industry-Grade Self-Sovereign Identity (IG-SSI) framework. Firstly, the preliminaries required for the framework are discussed. Next, the primary components are discussed and, finally, the secondary components.

B.1. Preliminaries

B.1.1. Attestations

Attestations can be said to be the core concept of Self-Sovereign Identity. With attestations, we refer to cryptographically signed data, enabling verification of information through validation of signatures. In other words, a client, i.e. an Authority, cryptographically signs—attests to—information for another party, the Subject. Allowing any third-party, a Verifier, to verify that the data was attested to by the Authority. As becomes apparent from this description, these roles are neither mutually exclusive nor static: a single party can both be e.g. the Authority and the Subject for an attestation, whilst being solely a Verifier in another instance. In IG-SSI, we make a distinction between three types of constructs which all may fall under the definition of attestations. The relationships between these constructs are visualised in Figure B.1 todoIncorporate different "attestation“ types into text body

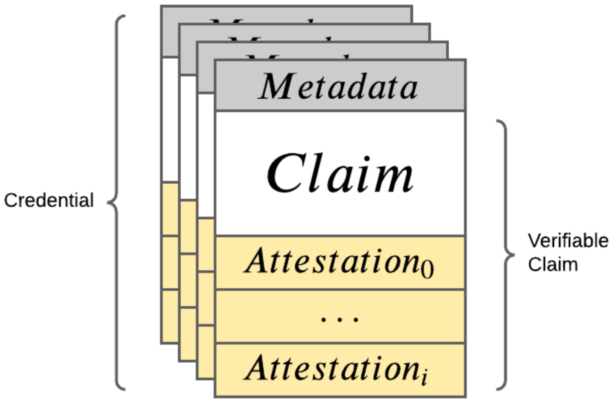


Figure B.1: The relationship between credentials and (verifiable) claims

Firstly, the entire construct of attestations relies on information or data, which is referred to as a claim. A claim comprises some information about a subject. A claim itself is relatively insignificant, as it would be in real life. Hence, in order to make claims verifiable and, therefore, trustworthy, attestations are introduced. An attestation is made for a certain claim, allowing other parties to generate trust in the claim through an attestation made by another party. This makes the claim—together with the attestation(s)—a verifiable claim as verifiability is achieved through the attestations. The same definition of a verifiable claim is proposed by Mühle et al., 2018. The information may be stored using a Zero-Knowledge Proof in order to control the privacy of the claim. Furthermore, specific information about a claim (e.g. the date of creation) may be stored in metadata. We deem metadata to be the distinguisher between a verifiable claim and an attribute as the metadata allows for the specification of certain properties the claim is bound by. An attribute can be deemed a defining characteristic of one's digital identity. Finally, authentication may require a selection of attributes, which we refer to as a credential. A credential does not need to be comprised of a single verifiable claim, e.g. a credential of one's identity may be composed of a claim of date of birth, country of residence, and one's legal name.

Asymmetric Encryption

In a centralised fashion, SSL-certificates are achieved through Public Key Infrastructure (PKI). In this infrastructure, a central authority can be said to attest to the identity of a domain, issuing a certificate, allowing verification of the domain through the signature of said authority. It is to note that this power of the authority can be delegated to it by a so-called root authority. An issuing authority is referred to as a Certificate Authority (CA). This relationships between CAs and domains can be compared to the issues being solved with SSI. In that sense, SSI can be compared to a Distributed PKI as described by Allen et al., 2015; Fromknecht and Yakoubov, 2014. PKI utilises public key encryption or asymmetric encryption. In public key encryption, any party possesses a public/private key-pair. The public key PK is considered public information, whilst the private key SK is never to be disclosed. Both these keys can be used for encryption, enabling different modus operandi. A party can encrypt a plaintext message m using his SK, creating ciphertext $c = encrypt(SK, m)$. Now, anyone possessing his PK has the possibility to decrypt c recovering the plaintext trough $m = decrypt(PK, c)$. This allows any decrypting party to ensure that the ciphertext was created by the party possessing the SK belonging to the PK. Vice versa, any party can encrypt m using the PK of another party, creating the ciphertext $c = encrypt(PK, m)$. Now, solely the owner of the SK belonging to the PK can decrypt the ciphertext, resulting in $m = decrypt(SK, m)$. This allows the encrypting party to ensure that only those with access to the SK can recover the contents of the ciphertext. An example of an asymmetric encryption scheme is Rivest et al., 1978

A rather straightforward realisation of Attestations can be achieved through asymmetric encryption and signatures. For instance an Authority can, through the use of his SK, encrypt the hash of a plaintext message (m) and the public key of the Subject (PK), resulting in $e(\mathscr{H}(m|pk))$. Encrypting the hash in this way is referred to as a signature. This allows any party that possesses the corresponding PK of the Authority, to verify that the data m was attested to by the Authority for the Subject. However, there exist several limitations with this approach. Firstly, disclosing the attestation and, thus, verifying the signature always reveals the corresponding plaintext values. This is not desirable, as the attestation may comprise sensitive data. Secondly, this would disclose more information than is necessary. For instance, verifying whether one is of age of majority, should not require the disclosure one's actual age. Rather, proving that one is above said threshold should suffice in such an instance. As such, Zero-Knowledge Proofs (ZKPs) may prove to overcome these hurdles.

Zero-Knowledge Proofs

ZKPs allow the verification of a value without disclosing the value to the Verifier Smart, 2016. ZKPs especially enable the integration of the minimisation property of IG-SSI. Three types of Zero-Knowledge Proofs can be distinguished: (1) exact proofs (ZKEPs) and (2) range proofs (ZKRPs), and (3) set membership proofs (ZKSM). Each ZKP can have interactive or non-interactive variants.

The general concept behind ZKPs is the ability for one to prove to another party that they know a specific value without disclosing said value. In ZKEPs, this value is exact. This could for instance be a proof referencing that one studies at TU Delft. Rather than actually containing the value, ZKPs store an assertion through which the knowledge of the plaintext value can be proofed. ZKRP following the same structure, however, they allow for assertion of knowledge in a range of values. For instance, proving that one's age is in between 18 and 200. This aids can both increase privacy or deteriorate it as in a ZKEP the verifying party must also be aware of the plaintext value in order to verify the knowledge of plaintext by the presenter. However, in a range proof the verifier must merely be aware of the range of values, greatly decreasing the search space. ZKRP do not reveal the actual value, however, hence they increase privacy through said search space. Finally, ZKSMs are similar to ZKRPs in the sense that they prove that a value is present in a specific set. Wherein ZKRPs this set is a range of integers, ZKSM allow any type of set. For instance, this enables proofs verifying that one's country of residence is part of the European Union or that one's function title is eligible for certain access roles.

Zero-Knowledge Exact Proofs

Boneh et al., 2005 proposes a ZKEP through the use of 2-DNF formulae over individual bits.

We propose the usage of ZKPs in IG-SSI for their added benefits of non-disclosure and range proofs. For regular static values exact proofs should be used, whilst any form of attestation requiring a number, range proof should be used.

Continue with info already written in

B.2. Architecture

B.2.1. Required Functionalities

An SSI scheme requires certain interactions in order to realise the asserted properties. Note that these interactions are a bare necessity, as more interactions allow for more fine-tuning of communication between parties. More specifically, we draw the distinction between primary and secondary issues. The following we regard as primary issues:

1. **Attestation Signing**
   This interactions comprises multiple states. An attestation is to be made over a piece of information. This is deemed a claim. When a claim is attested to by another party, it becomes verifiable, hence, creating a verifiable claim. A claim must already be verifiable to a certain extend as it contains information. In order to built trust in this information, it must be of integrity. Hence the entire stack of forming verifiable claims is a necessity for realising an SSI implementation.

2. **Attestation Verification**
   The verifiability of verifiable claims is another required component. As verifiable claims would be worthless without the means of verifying the actual data. This interactions enables the authentication of claims. This requires a certain degree of disclosure of underlying information of the claim. As discussed in section A.5, this must be minimised and only performed with justifiable parties.

3. **Attestation Presentation**
   The presentation of verifiable claims and, thus, attestations must be possible in order to allow verification and authentication of data. Again, this must be minimised and performed selectively in order to safeguard privacy.

4. **Revocation**
   An often undervalued and overlooked mechanic is that of revocation. Revocation must be performed in case a credential becomes invalid before the ending of its validity term has been reached. Revocation is focused on specifically in this thesis.

These four primary issues result in functionalities which can be regarded as the bare minimum for a functioning SSI architecture. Next, we discuss the secondary issues:

1. **Loss Recovery**
   Loss can occur in two-fold. We distinguish a loss of private key and a loss of attestations. Proper mechanisms must be implemented in order to aid recovery of these assets.

2. **Theft Locking**
   As storage is handled by the owner of the credentials, additional security consideration come into the equation. The owner must be properly secured against theft through both physical and virtual means.

The remainder of this chapter discusses the design of each of these core concepts.

B.3. Attestation Signing

As discussed previously, attestations are a concept which is comprised of multiple components. As such, IG-SSI draws similar distinctions. Primarily the distinction between claims and credentials. The design builds upon the work set out by Stokkink and Pouwelse, 2018 and Stokkink et al., 2020.

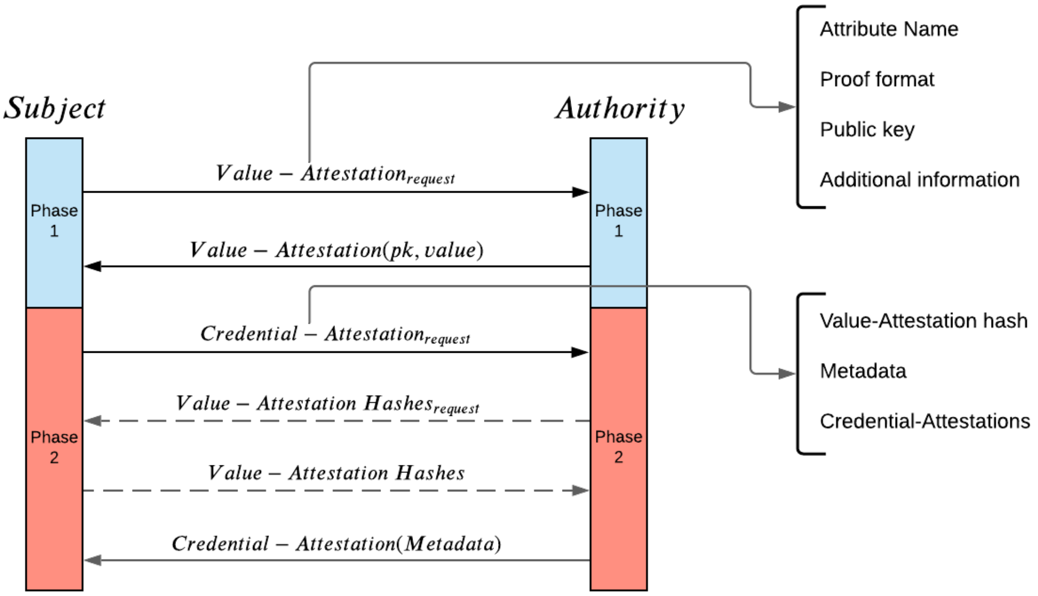

Subject
Authority
Phase 1
Phase 1
Phase 2
Phase 2
$Value - Attestation_{request}$
$Value - Attestation(pk, value)$
$Credential - Attestation_{request}$
$Value - Attestation\ Hashes_{request}$
$Value - Attestation\ Hashes$
$Credential - Attestation(Metadata)$
Attribute Name
Proof format
Public key
Additional information
Value-Attestation hash
Metadata
Credential-Attestations

Figure B.2: Attestation Flow

The distinction between two data structures is made for the attestation flow:

1. Claim: this structure can be said to be the core type. It is responsible for incorporating a specific value into a Zero-Knowledge Proof. The verifiable-nature of attestations stems from this type. As visible in Figure B.2, the design of this attestation allows for multiple proof formats, allowing for flexible selection of ZKPs and, thus, attestations. This disallows the lock-in of specific proof types, as any client can propose the usage of any type of proof, which can be used as long as the corresponding Authority supports the proposed type as well.

2. Credential: this structure is built around a claim. This type contains the actual attestation and allows for the subsequent attesting of values through attestation chaining: multiple authorities can attest for the same claim by attesting to the same claim as opposed to requiring a separate claim. Credentials also refer to metadata, which allows for validity terms and sign dates.

There exist several benefits to this construct. Firstly, the aforementioned chaining of attestations allows for multiple authorities to attest to a value. As such, real-life signature scenarios can be modelled through attestations. This allows for concepts such as segregation of duties and other shared responsibility scenarios, in which multiple parties must attest for a certain claim in order to be valid. For instance, a credential attesting for the ownership of a driving license, may require a signature by both a government body handing motor vehicles and a local government. The ability for multiple attestations for a single value can prove to be capable of handling such real-life scenarios. Secondly, subsequent attestations do not require the knowledge of the plaintext value. For instance, continuing on the driving license example, a local government does not require extensive knowledge on the license itself, a signature by the responsible government body should be enough for them to attest. As a consequence, this aids in data minimisation on subsequent Authorities. Finally, in case of attestation properties such as validity terms, a renewal of an attestation can simply be a new Credential-Attestation for the Value-Attestation, not requiring the re-attestation for the actual data. Again, this aids in data minimisation and privacy, as the plain text values do not have to be disclosed. Additionally, different Authorities can adhere to different metadata of the same attestation without influencing other parties. By allowing different Credential-Attestations for the same Value-Attestation, different metadata is enabled to exist for the same Value-Attestation. For instance, different Authorities can set different expiration dates on the same Value-Attestation. Again, when the expiration date has passed, the issuing Authority can simply re-attest for the same Value-Attestation, generating a new signature for the Credential-Attestation.

B.3.1. Attestation Flow

The attestation flow consists of two phases, the Proof-phase and the Credential-phase which do not always require subsequent execution. More specifically, for a single to be attested claim, the Proof-phase requires a single execution, which must occur before the Attestation-phase. Whilst subsequently, the Credential-phase can be performed indefinitely.

Claim-phase

The Proof-phase is initiated by a Subject. A Subject aims to have a claim attested to by an Authority. It does so by requesting an Attestation from the Authority. In this request, the Subject must make the attribute name, the to be used proof format, and his public key apparent. This public key, is a one time used public key, of which the private key must be stored by the Subject. The usage of single-use public/private key pairs, allows for additional privacy properties imposed on the system, which will be explained in. Additionally, any other information that is to be known by the Authority must be sent along, for instance the requested plaintext value. Note that the value is, thus, not required to be sent by the Subject. The implication of this, is that an Attestation can be made for the Subject, without the Subject knowing the exact value. This, hence, allows for the secure storage of information, in the form of a ZKP attestation on a client, without the actual revealment of the underlying value.

ence + write small
n this subject

The receiving Authority may respond to the request, making him an issuing Authority. The Authority generates a Value-Attestation of the type defined by the proof format. This attestation, thus, incorporates the value belonging to the requesting attribute name. This attestation is sent back to the requesting Subject. After having received the Value-Attestation, the requesting Subject moves onto the Credential-phase.

Whilst the main focus of this interaction relies on a separate entity in order to create the claim using a ZKP, this is not required. A client may self-sign an claim, hence, having full control of the incorporated value. However, from a usability perspective and in sight of passport-grade credentials, the ability for other clients to determine the value which is imposed upon the claim, is desirable. For instance a client may have no knowledge of the claim that is to be made. For instance, due to a specific structure required for a claim. In such an instance it would be cumbersome to have an Authority firstly make the correct value apparent and afterwards having said Authority attest to the claim. This would introduce additional overhead to the protocol. Similarly, an Authority may want to ensure that a specific value is being used, without requiring any verification afterwards.

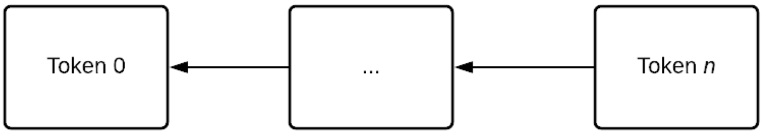


Figure B.3: Token Chain

Credential-phase

In the Credential-phase, a requesting Subject requests an attestation for a certain Value-Attestation, making it a Credential. It does so by disclosing all already attested Credential-Attestations belonging to the Credential. The core of each Credential is an Attestation Token. Each Token contains the hash of a Value-Attestation and points to the previous Token. This has been visualised in Figure B.3. The first token, comparable to a genesis-block in Blockchain structures such as that by Nakamoto, 2009, contains the hash of the public key belonging to the Subject. Any subsequent Credential, thus, generates a new Token, occupying a place as a shackle in the chain. When an Authority is requested to attest to a Credential, it may request each previous token and, thus, the hashes of each previous attestation, after which it can verify these attestations. As such, it is improbable for a client to attempt to hide the existence of an attestation or attempt to cheat the system, as otherwise the attestations of other Authorities become invalid (as the hash of the token will no longer be correct). Hence, as visible in the second phase of in Figure B.2, after having received a Credential request, the Authority may request any missing tokens until he gains confidence to attest for the Credential, creating a Credential-Attestation. Note that these Tokens do not reveal any information about the underlying Value-Attestations, as they merely contain the hash value. When an Authority attests to a Credential, it generates a signature for the hash of the corresponding metadata, which in turn points to a Token. This structure of referencing data structures is visualised in Figure B.4.

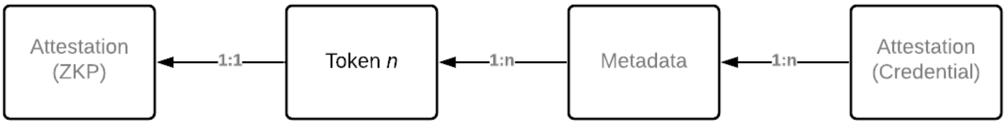


Figure B.4: Data Structure Relationships

As portrayed, a Token refers to a single Value-Attestation. However, multiple metadata instances may reference a single Token and, similarly, multiple Credential-Attestation may reference a single metadata instance. These relationships allow for the aforementioned properties and scenarios. As becomes apparent from this description, the second phase, i.e., the Credential-Phase, can thus be repeated indefinitely as numerous Authorities can co-attest and re-attest for an Attestation.

B.4. Attestation Presentation

n requests

In order to verify attestation values, a presentation procedure must exist. As clients may decide themselves whether to share attributes, we propose the structure as visible in Figure B.5 . In this structure, an Authority requests an attribute with a specific name. A Subject may subsequently decide whether to respond to such a request and to disclosure the corresponding attribute. Note here that the credential request is not necessarily required, as a client can disclosure an attribute directly. However, the specification of an attribute name aids in selective disclosure, whilst additionally allowing the Authority to determine whether a specific credential is solicited. Furthermore, in order to aid distinguishability between outstanding requests, a Verifier may disclose a nonce which is to be sent back by the Subject. Through this nonce, After a credential has been disclosed and, thus, presented, the Authority may verify its validity.

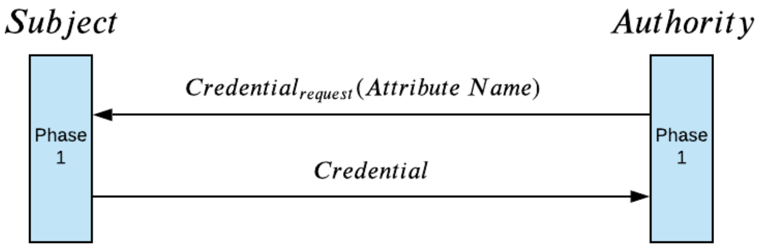


Figure B.5: Attestation Presentation

B.5. Attestation Verification

Verification

We propose two types of verification. Firstly, an interactive variant and, secondly, a non-interactive variant, enabling offline verification. The general flow of the interactive variant is visible in Figure B.6. For active verification, an Authority requests the underlying Value-Attestation by presenting the attestation hash to the Subject. The Subject may consent through sending the requested Attestation. After the Authority receives the ZKP commitment, the Authority may send challenges to verify the underlying value. Note that for this to happen, the Authority must either be already aware of the value belonging to the attribute or the plaintext value must be shared. Sharing of the plaintext value can be done during presentation-time. This should be performed using encryption in order to preserve privacy, for instance through the use of RSA by Rivest et al., 1978.

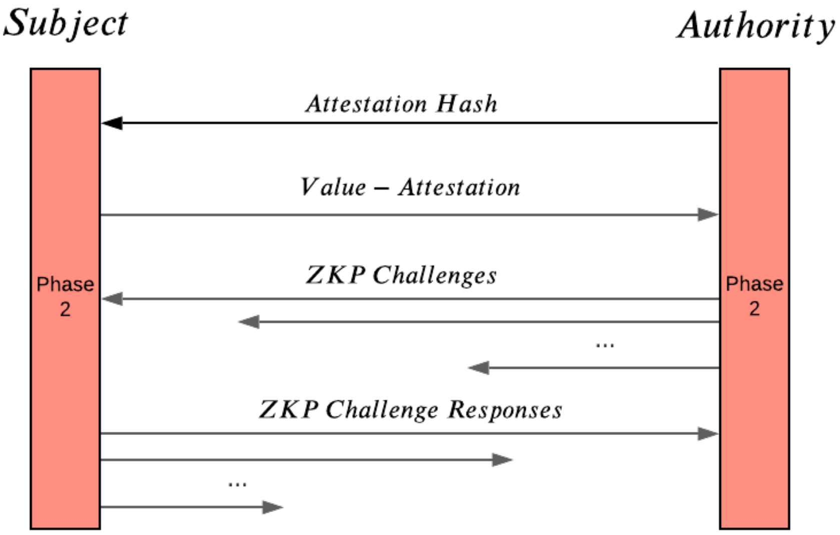


Figure B.6: Interactive Verification

The second method for verification uses the attestations made by other authorities. In order for this attestation to pass, the list of attestors must contain an authority that is trusted by the Verifier. If this is the case, a Verifier may accept the value proposed by the Subject in case the metadata contains the hash of this value and the signature made by one of the acknowledged authorities over the metadata is valid. This approach does not require any connectivity between the Subject and Verifier, apart from the presentation itself. However, a presentation does not necessarily require any form of digital communication (e.g. through QR-codes). It is, however, to note that this offline verification, thus, does not rely on any additional token requests and, as such, all tokens must either be made directly apparent to the Verifier during presentation-time or the verifier must make its decision based on the presented Attestation and his reliance on and knowledge of acknowledged authorities.

B.6. Attestation Revocation

Revocation is one of the main unsolved issues in Self-Sovereign Identity and an issue in distributed systems as a whole. As in real life contracts and other agreements may become invalid before their termination date, the ability to revoke attestation in SSI must be available as well. Several motivations exist for revocation:

- Erroneously signed data: in case data was signed accidentally.
- A Legally invalid contract: in case at a later instance it became apparent that the signed data can not be legally upheld.
- Premature termination of a contract: in case a certain breach of contract occurs.

Note that expiration is not one of these listed motivations, as time-bound attestations can be realised using signed metadata. It is important that revocation can never occur due to expiration, as some claims should never be able to be revoked. For instance, it should not be possible for an authority to revoke a signature indicating someone is of legal age (unless in the rare instance that it was erroneously signed and can be publicly verified that this was, indeed, the case), as this fact can never become false.

As IG-SSI is built without specialised validation nodes, present in some blockchain-based protocol such as Zhou et al., 2019, there is no trivial non-interactive solution of revocation of signatures. The trivial solution is to actively query signees (i.e., the responsible authorities) and verify that they still attest for the signed information. There exist multiple problems with this solution. Firstly, this querying requires interactivity with the signee(s) of an attestation. Whilst interactivity is not a problem per se, it does introduce additional overhead. It requires the signee(s) to be online. Whilst availability often is a key characteristic in distributed systems, there is no guarantee that specific clients, i.e. the Authorities, are available. Additionally, this interactivity generates overhead in the verification process: apart from challenging the presenting client, the signees have to be actively queried, introducing additional verification time and network traffic.

Secondly, as a requirement for enabling this interactivity, a (network) connection to the signees must be available. This completely nullifies the possibility for offline verification. Next, we discuss our solution for revocation: The Hybrid Revocation Model (HRM). This model requires no additional interactivity during verification and enables offline-verification.

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, 2017, 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 et al., 2009 or the usage of accumulators as described by Camenisch et al., 2010; 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 et al., 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[1] and more than 200GB[2] for regular and 4TB[3] 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.

B.6.1. Remove?

remove?

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 B.7, 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.

B.7. 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 B.8. 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.

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

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

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

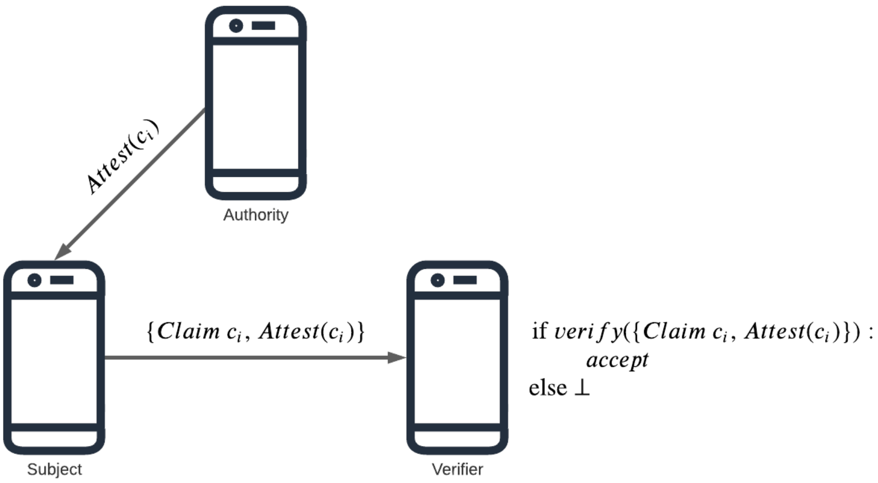

$Attest(c_i)$
Authority
Subject
$\{Claim\ c_i,\ Attest(c_i)\}$
Verifier
if $verify(\{Claim\ c_i,\ Attest(c_i)\})$:
$accept$
else $\perp$

Figure B.7: Revocation requiring interactivity

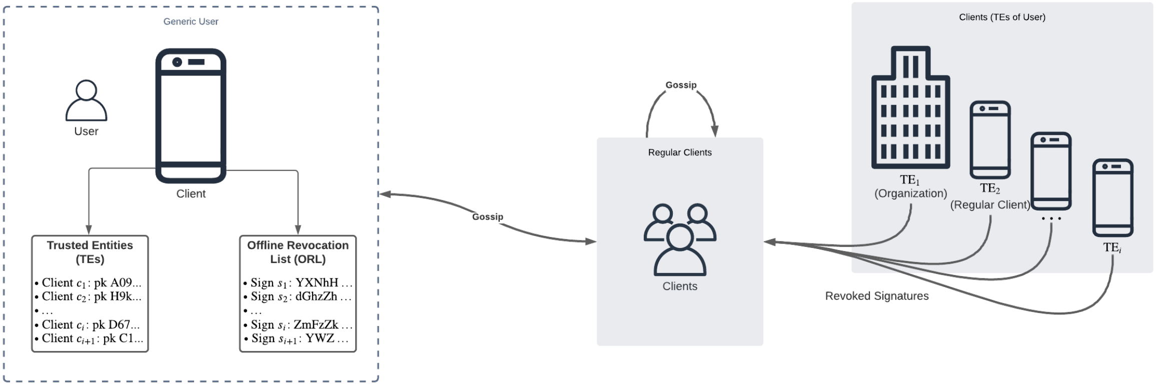


Figure B.8: Hybrid-Revocation Model (HRM)

Hybrid Revocation Model

The Hybrid Revocation Model attempts to overcome the hurdle of interactivity whilst allowing for flexibility, enabling offline-verification. IG-SSI is fully distributed and as such, each node is equal. As a consequence, the client performing the verification must be aware of any revocations belonging to a presented attestation. Selecting specific nodes for distributing and holding revocation, would deteriorate the equality principle. As these nodes would, then, possess the ability to hide certain revocations from the network or could lead to collusion (Khovratovich & Law, 2017). As such, revocations should be public data. I.e., every revocation should be visible to every client. The hybrid nature of the model, stems for its offline capabilities: during verification-time, clients do not require to be online. They merely require occasional synchronisation of revoked attestations through communication with other peers.

In HRM, each peer has the possibility to posses the same information about revocations. Revocations are propagated through the network, enabling each peer to store revocations from clients they trust. This concept builds upon the notion of Trusted Authorities. The general flow of the design can be seen in Figure B.8. The protocol has three key concepts:

1. Trusted Authorities (TAs)
2. Propagation
3. Offline Revocation List (ORL)

Next, we explain each concept.

Trusted Authorities

In a fully distributed setting, client are responsible for their own actions. Meaning that revocation are as meaningful as the extend to which they are used by the clients. This property makes it that clients themselves are able to acknowledge or reject revocations. A criterion on which a client is able to determine the validity of a revocation is whether the Revoking Authority is trusted by the client. This is where we introduce the notion of Trusted Authorities (TAs). As mirrored by real life, a person has (relatively speaking) a choice whether to acknowledge a certain authority. With SSI aiming to be a digital extension to one's identity, one should also be able to make such an acknowledgement in the digital domain. As an added benefit, identification in the digital domain can prove to be more verifiable than physical verification. We propose the usage of a Trusted Authority Storage (TAS). In the TAS, the public key and the public key hash of a TA are stored. We make the distinction between acknowledged (trusted) and Unacknowledged Authorities (UAs). As discussed previously, client roles are neither static nor mutually exclusive. As a consequence, potentially every client can be an Authority. However, it is up to a client to determine whether an authority is a TA or an UA. In terms of distributed revocation: a client aims to accept only those revocations of which he knows that he can trust the authenticity. The results of acceptance are the storage of the revoked signatures and propagation towards network.

Propagation

In order to safeguard availability in the network and enable offline verification, we propose the propagation of revocations throughout the network. This requires two means: firstly, a verifiable revocation format and, secondly, a propagation protocol for the revocations. We propose the structure as visible in Table B.1. This design, in addition to the revoked hashes, includes a public key hash, a version number, a specification for the used hashing algorithm and a signature. The public key hash allows for the retrieval of the public key in case said key belongs to a TA acknowledged by the receiving client. This public key can, thus, be retrieved by querying the TAS. In case the public key belongs to a TA, the signature can be verified by concatenating the version number with the revocations. Unique version numbers allow clients to ensure that they are either fully synced with the network or are missing certain revocation versions. The revocations themselves are to be the hashes belonging to the attestation metadata. This, thus, invalidates any attestations made to this metadata and the token it points to. As a benefit, this reduces overhead when presenting attestations as solely based on the metadata, an attestation can be deemed to be valid or revoked. The hashing algorithm specification improves the transparency and robustness of the schema. For instance, hashing algorithm recommendation may differ in the future due to e.g. efficient collision finding. Allowing for specification enables the interchanging of this algorithm, aiding future-proofness and flexibility.

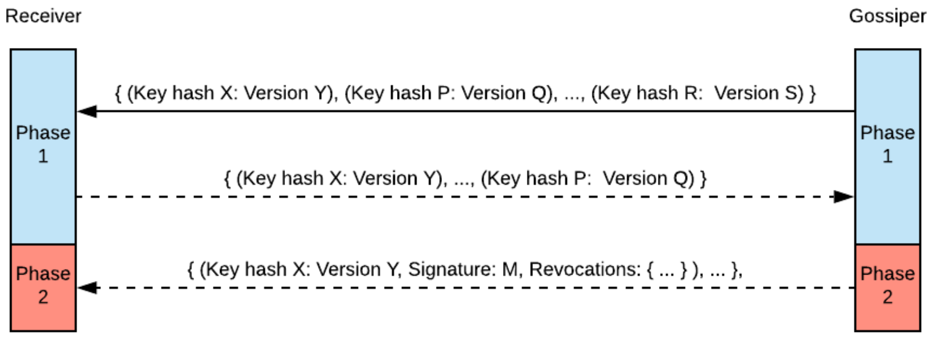

Receiver
Gossiper
Phase 1
Phase 1
{ (Key hash X: Version Y), (Key hash P: Version Q), ..., (Key hash R: Version S) }
{ (Key hash X: Version Y), ..., (Key hash P: Version Q) }
Phase 2
Phase 2
{ (Key hash X: Version Y, Signature: M, Revocations: { ... }), ... },

Figure B.9: Multi-step Update Procedure

The propagation itself requires a protocol that ensures information is (eventually) spread across the entire network, whilst also ensuring that unavailable nodes receive the information at a later instance. For this, we propose the usage of gossip protocols with interval re-transmission. Gossip protocols are communication protocols which allow for the periodic exchange of data with (random) peers (Kwiatkowska et al., 2008).

The periodic exchange of data with peers, makes gossip protocols a prime candidate for the realisation of distributed revocation. Furthermore, in order to decrease the overhead of gossiping a theoretically unbound number of signatures, we propose the usage of a multi-step update procedure. This procedure has been visualised in Figure B.9. This procedure is split-up in two phases: firstly, a gossiping client gives notice to a client that it possesses specific authority-version pairs, containing the public key hash of an authority and the latest version it is aware of. Next, the receiving client can request an update by sending back the latest versions of the revocations stored in their TAS. This allows a client to selectively send updates, as the receiving party makes an underbound of the known versions apparent. This extra step of selective requesting relieves a large amount of data as clients are not necessarily interested in revocations by certain authorities as they may be considered UAs or a client may already be fully synced.

We note that this procedure may be fine-tuned through the usage of revocation dates. Revocation dates may allow clients to opt out of old revocation versions, optimising storage usage as old revocations may no longer be relevant in the system due to the validity terms of the attestations having passed.

Offline Revocation List

Any valid received revocation should be stored by a client for later reference. The storage of revocations allow for offline (in)validation of attestations. This storage we deem the Offline Revocation List (ORL). Whilst no specific storage structure is required, we do propose the usage of Bloom filters for member checking. A Bloom filter is a memory- and time-efficient probabilistic data structure, which allow for efficient membership operations (Bloom, 1970). Raya et al., 2007; Raya et al., 2006 discuss the benefits of Bloom filters in Certificate Revocation Lists (CRLs), which can be transformed to our concept of ORL, as the ORL can be deemed a more generic variant of a CRL.

As yearly up to 340.000 identity documents are stolen in a country as The Netherlands, the same amount of revocations must be possible on a year basis (Nieuwsuur, 2019). As such, revocation membership checking can prove to become quite expensive both memory- and runtime-wise. Even with the most efficient algorithms such as Binary search, with a runtime complexity of $\mathcal{O}(\log(n)$, the execution time of such a search can be too long, usability-wise. As such, we propose the usage of membership verification through Bloom filters, in which a membership search on the actual data is only performed in case of a possible match. Additionally, it can be said that the probability of encountering a revoked attestation should be extremely unlikely. As we assume the majority of the nodes to be honest, they have no incentive to attempt to cheat the system. As such, Bloom filters with their property of ensuring an item has no membership in case the filter does not contain it and, thus, only having to validate using the actual data in case the filter may contain the item, Bloom filter can prove to achieve much stricter execution timings for validation.

Furthermore, we note that the ORL can be replaced by a Bloom filter entirely. A client may chose to accept the probabilistic nature of Bloom filters over the exact membership check from memory. Such nodes may not be able to aid in the propagation of the revocations, however, the low memory requirements may prove to make the protocol suitable for IoT devices.

Table B.1: Verifiable Revocation Update Format

Authority key hash	5e2bf57d3f40c4b6df6...
Version	1701
Hashing Algorithm	SHA3-256
Signature	422c06fbb4fbd23d33...
Revocations	b788c5b28dba2fc6a0... 7f2519609cf157d7e9... ... e2d7610dcb53724675...

C
Analysis & Results

This section outlines the results from the proposed design discussed in **??**. Implementation details are discussed, testing methodologies, as well as theoretical analyses on runtimes.

C.1. Results

C.2. Runtime Analysis

C.2.1. Synchronisation Upperbound

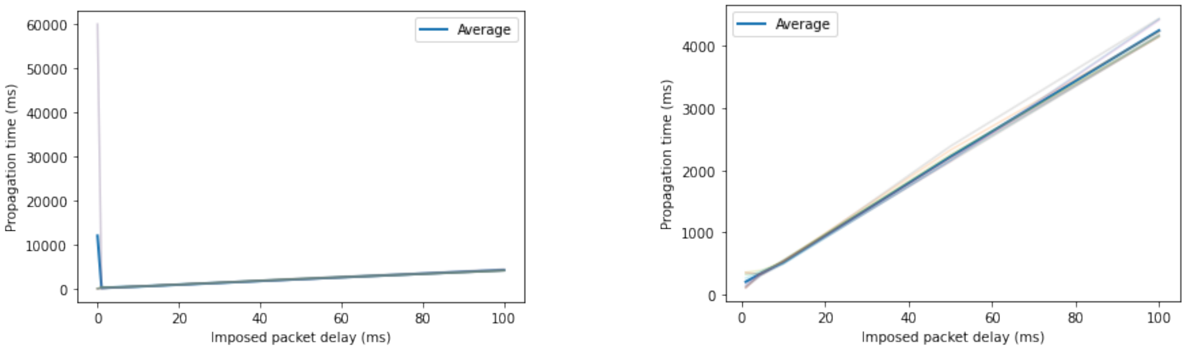
Synchronisation in the system is dependent on the entire community of peers. Whilst consensus on revoked signatures is reached on peer-level, propagation is dependent on the entirety of peers. I.e., revocations are sent across the network in a peer-to-peer fashion. More specifically, peers are to actively 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:

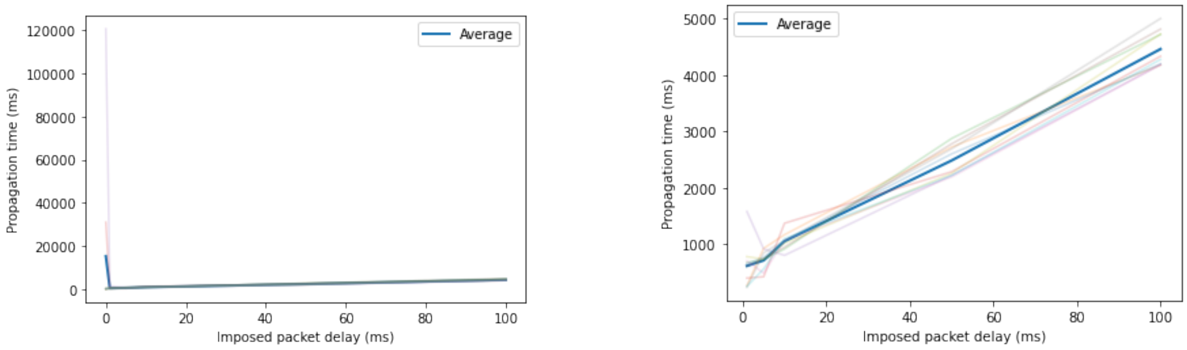
1. **Gossip-interval** (t_g): the time interval on which peers are gossiped to.



(a) UDP Impact

(b) UDP Impact (highlighted)

Figure C.1: UDP Impact ($n = 1, m = 1$)



(a) UDP Impact

(b) UDP Impact (highlighted)

Figure C.2: UDP Impact ($n = 1, m = 10$)

2. **Gossip amount** (n_g): the number of peers which are gossiped to on a time interval.
3. **Peer selection** ($\mathscr{F}_g(\mathscr{X})$): 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 $g = t_g \cdot \dfrac{n_p}{n_g}$ be the minimal number of interval iterations required to gossip to all peers. The peer selection function $\mathscr{F}_g(X)$ 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 $m_p >= n_p$ selected throughout each iteration until convergence. I.e., the peer selection function $\mathscr{F}_g(X)$ selected at least $m_p >= n_p$ peers, leading to at least $t_g \cdot \dfrac{m_p}{n_g}$ iterations. The additional iterations can be modelled by: $h = t_g \cdot \dfrac{m_p - n_p}{n_g}$, 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 C.1.

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

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 C.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.

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

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 $\mathscr{F}$ allows for multiple modus operandi. E.g., the $\mathscr{F}$ 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 $\mathscr{T}$. 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 $\mathscr{D} = \{\delta(p_0), \ldots, \delta(p_{n-1})\}$ be the set containing all these smallest propagation delays for each node. Finally, let $\mathscr{C} = \{c_0, \ldots, c_{n-1}\}$ 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 network with size n as summarised in Equation C.3

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

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 C.4.

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

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 Equation C.5

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

The propagation time for a network of size n, is visible in Equation C.6

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

Bibliography

Allen, C., Brock, A., Buterin, V., Callas, J., Dorje, D., Lundkvist, C., Kravchenko, P., Nelson, J., Reed, D., Sabadello, M., et al. (2015). Decentralized public key infrastructure. a white paper from rebooting the web of trust.

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

Aristotle. (1925). Metaphysics (W. Ross, Ed.; T. I. C. Archive, Trans.) [Original work published 350 B.C.E.]. http://classics.mit.edu/Aristotle/metaphysics.4.iv.html

Baars, D. (2016). Towards self-sovereign identity using blockchain technology.

Belchior, R., Putz, B., Pernul, G., Correia, M., Vasconcelos, A., & Guerreiro, S. (2020). SSIBAC: Self-Sovereign Identity Based Access Control (tech. rep.). https://vonx.io/

Bertino, E. (2006). Establishing and protecting digital identity in federation systems. Article in Journal of Computer Security. https://doi.org/10.1145/1102486.1102489

Bloom, B. H. (1970). Space/time trade-offs in hash coding with allowable errors. Communications of the ACM, 13(7), 422–426.

Boneh, D., Goh, E. J., & Nissim, K. (2005). Evaluating 2-DNF formulas on ciphertexts. Lecture Notes in Computer Science, 3378, 325–341. https://doi.org/10.1007/978-3-540-30576-7{_}18

Braendgaard, P. (2017). What is a uPort identity? https://medium.com/uport/what-is-a-uport-identity-b790b065809c

Camenisch, J., Kohlweiss, M., & Soriente, C. (2010). Solving revocation with efficient update of anonymous credentials. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 6280 LNCS, 454–471. https://doi.org/10.1007/978-3-642-15317-4{_}28

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.

Cameron, K. (2018). Let's find a more accurate term than 'Self-Sovereign Identity'. https://www.identityblog.com/?p=1693

Camp, L. J. (2004). Digital Identity. https://doi.org/10.1109/MTAS.2004.1337889

Comission, E. (2021). Regulation of the european parliament and of the council amending regulation (eu) no 910/2014 as regards establishing a framework for a european digital identity.

Council, P. S. S. (2004). Payment Card Industry Data Security Standard (PCI DSS).

Decentralized Identifiers (DIDs) v1.0. (n.d.). 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.

Fromknecht, C., & Yakoubov, S. (2014). A Decentralized Public Key Infrastructure with Identity Retention (tech. rep.).

Good ID. (2021). Glossary. https://www.good-id.org/en/glossary/self-sovereignty/

Hall, B. K., Benedikt, H., & Strickberger, M. W. (2008). Strickberger's evolution. Strickberger's evolution (4th ed., pp. 4–4). Jones & Bartlett Learning.

IANA. (n.d.). IANA — Number Resources. https://www.iana.org/numbers

IBM. (2019). Consumer Attitudes Towards Data Privacy.

IBM. (2021). Identification and authentication - IBM Documentation. https://www.ibm.com/docs/en/ibm-mq/9.1?topic=mechanisms-identification-authentication

ICANN. (2017). https://www.icann.org/resources/pages/register-domain-name-2017-06-20-en

ISO. (2019). IT Security and Privacy — A framework for identity management — Part 1: Terminology and concepts (Standard). International Organization for Standardization.

ISO. (2013). ISO/IEC 27001 INFORMATION SECURITY MANAGEMENT (tech. rep.). International Organization for Standardization.

Jøsang, A., & Pope, S. (2005). User Centric Identity Management. AusCERT Conference 2005. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.60.1563&rep=rep1&type=pdf

Khovratovich, D., & Law, J. (2017). Sovrin: digital identities in the blockchain era (tech. rep.). http://www.credentica.com/the

Kwiatkowska, M., Norman, G., & Parker, D. (2008). Analysis of a Gossip Protocol in PRISM (tech. rep.). http://www.prismmodelchecker.org/casestudies/gossip.php

Lasla, N., Younis, M., Znaidi, W., & Ben Arbia, D. (2018). Efficient Distributed Admission and Revocation Using Blockchain for Cooperative ITS. 2018 9th IFIP International Conference on New Technologies, Mobility and Security, NTMS 2018 - Proceedings, 2018-January, 1–5. https://doi.org/10.1109/NTMS.2018.8328734

LastPass. (2019). THE 3RD ANNUAL GLOBAL PASSWORD SECURITY REPORT (tech. rep.). LastPass. https://lp.logmeininc.com/rs/677-XNU-203/images/LastPass_State-of-the-Password-Report.pdf

Loffreto, D. (2012). What is "Sovereign Source Authority"? https://www.moxytongue.com/2012/02/what-is-sovereign-source-authority.html

Loffreto, D. (2016). Self-Sovereign Identity. https://www.moxytongue.com/2016/02/self-sovereign-identity.html

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

Merriam Webster. (n.d.). Identity. Retrieved July 11, 2021, from https://www.merriam-webster.com/dictionary/identity

Moore, M. (2019). What is Industry 4.0? Everything you need to know. https://www.techradar.com/news/what-is-industry-40-everything-you-need-to-know

Morin, D. (2008). Announcing Facebook Connect. https://developers.facebook.com/blog/post/2008/05/09/announcing-facebook-connect/

Mühle, A., Grüner, A., Gayvoronskaya, T., & Meinel, C. (2018). A survey on essential components of a self-sovereign identity. https://doi.org/10.1016/j.cosrev.2018.10.002

Nakamoto, S. (2009). Bitcoin: A Peer-to-Peer Electronic Cash System (tech. rep.). www.bitcoin.org

Nakanishi, T., Fujii, H., Hira, Y., & Funabiki, N. (2009). Revocable Group Signature Schemes with Constant Costs for Signing and Verifying (tech. rep.).

Nieuwsuur. (2019). We verliezen massaal onze paspoorten: fraudemeldingen verdubbeld | Nieuwsuur. https://nos.nl/nieuwsuur/artikel/2292728-we-verliezen-massaal-onze-paspoorten-fraudemeldingen-verdubbeld

Noonan, H., & Curtis, B. (2018). Identity. In E. N. Zalta (Ed.), The stanford encyclopedia of philosophy (Summer 2018). Metaphysics Research Lab, Stanford University.

Olson, E. T. (2021). Personal Identity. In E. N. Zalta (Ed.), The stanford encyclopedia of philosophy (Spring 2021). Metaphysics Research Lab, Stanford University.

Othman, A., & Callahan, J. (2018). The Horcrux Protocol: A Method for Decentralized Biometric-based Self-sovereign Identity. Proceedings of the International Joint Conference on Neural Networks, 2018-July. https://doi.org/10.1109/IJCNN.2018.8489316

Philpott, D. (2020). Sovereignty. In E. N. Zalta (Ed.), The Stanford encyclopedia of philosophy (Fall 2020). Metaphysics Research Lab, Stanford University.

PressPass. (1999). Microsoft Passport: Streamlining Commerce and Communication on the Web. https://web.archive.org/web/20071214080959/https://www.microsoft.com/presspass/features/1999/10-11passport.mspx

Preukschat, A., & Reed, D. (2021). Self-sovereign identity: Decentralized digital identity and verifiable credentials. Manning Publications Co. LLC.

Raya, M., Papadimitratos, P., Aad, I., Jungels, D., & Hubaux, J.-P. (2007). Eviction of Misbehaving and Faulty Nodes in Vehicular Networks. IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, 25(8). https://doi.org/10.1109/JSAC.2007.0710xx

Raya, M., Jungels, D., Papadimitratos, P., Aad, I., & Hubaux, J.-P. (2006). Certificate Revocation in Vehicular Networks (tech. rep.). https://www.researchgate.net/publication/37433732

Recordon, D., & Reed, D. (2006). Openid 2.0: A platform for user-centric identity management. Proceedings of the Second ACM Workshop on Digital Identity Management, 11–16. https://doi.org/10.1145/1179529.1179532

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

Rivest, R. L., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 21(2), 120–126.

Rosenberg, M. (2018). Cambridge Analytica and Facebook: The Scandal and the Fallout So Far. https://www.nytimes.com/2018/04/04/us/politics/cambridge-analytica-scandal-fallout.html

Ruff, T. (2018). 7 Myths of Self-Sovereign Identity. https://medium.com/evernym/7-myths-of-self-sovereign-identity-67aea7416b1

Sheldrake, P. (2016). [On the misattribution in Allen (2016)]. https://sheldrake.medium.com/see-also-http-www-lifewithalacrity-com-2016-04-the-path-to-self-sovereign-identity-html-44c6b53cd737

Smart, N. P. (2016). Cryptography made simple (pp. 425–438). Springer.

Sovrin. (2019). Sovrin SSI & IoT Working Group Charter (Version 1).

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

Speelman, T. (2020). Self-Sovereign Identity: Proving Power over Legal Entities (Doctoral dissertation). TU Delft. http://resolver.tudelft.nl/uuid:aab1f3ff-da54-47f7-8998-847cb78322c8

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. 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), 1336–1342.

Thales. (2020). 2020 Thales Data Threat Report (tech. rep.). Thales. https://cpl.thalesgroup.com/sites/default/files/content/research_reports_white_papers/field_document/2020-04/2020-data-threat-report.pdf

The European Parliament and Council. (2016). Regulation (EU) 2016/679 of the european parliament and of the council. https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32016R0679%5C#d1e6226-1-1

The Oxford Dictionary. (n.d.). Govern.

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.).

Verizon. (2020). 2020 Data Breach Investigations Report (tech. rep.). Verizon. https://enterprise.verizon.com/resources/reports/2020-data-breach-investigations-report.pdf

Zhou, T., Li, X., & Zhao, H. (2019). EverSSDI: Blockchain-based framework for verification, authorisation and recovery of self-sovereign identity using smart contracts. International Journal of Computer Applications in Technology, 60(3), 281–295. https://doi.org/10.1504/IJCAT.2019.100300

Zimmermann, H. (1980). Osi reference model-the iso model of architecture for open systems interconnection. IEEE Transactions on communications, 28(4), 425–432.