




Obstacles In Implementing and Integrating Self-Sovereign Identity and Proposed Solutions

Erik Hieta-Aho , Anni Karinsalo , Valtteri Lipiäinen 

VTT, Technical Research Centre of Finland

e-mail: {erik.hieta-aho | anni.karinsalo | valtteri.lipiainen}@vtt.fi

Abstract—Self-Sovereign Identity is an approach to handling identity documents in a user-centric manner around user consent. Specifically, identity documents are always controlled by the user they identify, and any operations involving them are carried out by the user. This is in contrast to existing models of identity management where information is stored and processed in a centralized manner by an external party. This approach gives rise to many applications including service authentication, cross-border travel, education focused credentials, and other user-centric technological use cases. However, work is still required for these applications to become reality. One of the key issues is that a fragmented standards space hinders efficient product building. A potential consequence is the emergence of compartmentalized ecosystems, which would drastically decrease systems' user-friendliness. This paper lays out obstacles faced in implementing and integrating SSI, including interoperability issues, and suggests how the situation could be improved.

Keywords—Digital credentials; Identification; Decentralized; Standards; Interoperability.

I. INTRODUCTION

As online services become more and more integrated into daily life, concerns about security and privacy become more relevant. More often than not, a service requires authentication, be it for access to paid services or simply access to the history and data of the user. The traditional way of handling authentication, usernames and passwords, has serious flaws: having to use multiple passwords either leads to repetition (which poses a security threat), or makes accessing services cumbersome. Single Sign-On (SSO) offers a solution to this issue: a user only has to remember one password, and the corresponding identity can be used to authenticate to multiple services. A number of standards have been established in this area, including OpenID Connect. However, this approach still means that user data is stored by a centralized authority.

Self-Sovereign Identity (SSI) aims to improve on the SSO approach by making the user (whose identity data is being handled) sovereign over that data. Technically, this means authenticating using Verifiable Credentials (VCs), which contain user information. The credentials are stored by the user and only used at the user's discretion. In addition to plain user data, the credentials can also contain other sensitive information, such as travel documents, without the user having to relinquish control of them to any identity provider. This opens up the possibility for further applications in a privacy-preserving manner.

A natural question is how one can prove their identity in this setup, where the credentials are controlled by the user. The answer is twofold. First, VCs identify their owner

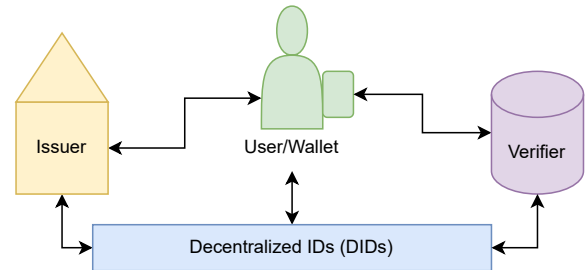


Figure 1. Self-Sovereign Identity

by use of a Decentralized Identifier (DID). A DID allows the creation of a persistent identity that is not linked to any centralized authority. There are multiple different DID methods for achieving this goal, from traditional public-key cryptography to blockchain-based solutions. Second, trusted issuers can vouch for the identity of a user. For instance, a government will only issue a passport VC to a user it has identified as the one holding the given passport. In comparison to other digital identity frameworks (e.g. government-issued eID solutions) SSI has the benefit of being decentralized and so the interoperability across borders may be considered less challenging.

The SSI approach is still relatively new, and the ecosystem is still taking shape. The unique benefits SSI can bring for users are only achievable if systems are interoperable, as described in Section III. Standards are a great starting point for interoperability, fortunately there is a rich standards ecosystem for SSI, as described in Section IV. Even though standards exist in this space, there are major areas of friction when attempting to integrate different systems, as we have faced in a project implementing SSI for authentication. The challenges mainly relate to the flexibility in encoding formats, and are described in detail in Section V. We believe that these obstacles may cause major issues for the establishment of an SSI ecosystem, and in Section VI we describe approaches, mostly related to clarity in metadata and standards texts, for lessening this impact.

II. SSI

The original concept for SSI is defined by [1] and further refined by [2]. Essentially, the following principles were defined:

- **Existence:** Users must have an independent existence beyond digital systems.
- **Control:** Users must control their identities and how they are used.
- **Access:** Users must have access to their own data without intermediaries.
- **Transparency:** Systems and algorithms managing identities must be open and transparent.
- **Persistence:** Identities should be long-lived and ideally last indefinitely.
- **Portability:** Identities should be transportable across different systems.
- **Interoperability:** Identities should work across various platforms and technologies.
- **Consent:** Users must agree to how their identity data is shared and used.
- **Minimization:** Identity systems should only collect and share necessary data.
- **Protection:** Users' rights and privacy must be safeguarded.

To fulfill this promise, a set of technologies have been proposed. For handling identity, the technologies of decentralized identifiers and verifiable credentials can be used.

VCS are objects that encode verifiable information about a user. Fundamental to the SSI approach is a separation of the actors into three distinct groups: wallets, issuers and verifiers, which can be seen in Figure 1. The wallet is where a user stores their credential. The issuer is the party that issues credentials to be stored in the user wallets. The verifier is then in charge of checking whether a given credential is valid. All three have to interact for the system to work.

An example of this setup in action is a university diploma and can be seen in Figure 2. After a student graduates, they may want to receive a digital diploma, which they can use in the future to prove they have graduated (for instance when applying for jobs, or applying for a further degree). Following an SSI approach, the student should wholly control how they use the diploma after it is granted: there should be no need to go back to the university each time (and for instance let them know what jobs the student is applying to). For this reason, the student should have an SSI wallet, which is independent of the university.

To receive a university diploma VC, the student would then interact with an issuer tied to the university, the party that is authorized to issue such diplomas. After the student has shown that they are indeed entitled to a credential, the issuer creates and signs a credential tied to that student (through their unique DID), and sends it to the user. After this point, the credential is controlled by the student. For the credential to be useful, anyone it is presented to should be able to confirm it was issued by the university to this person. This is where the verifier becomes relevant. They are in charge of verifying that the credential is valid, and that this student has indeed graduated from the university.

This sort of setup is useful for making it clear how the roles of the wallet, issuer and verifier are different entities. There

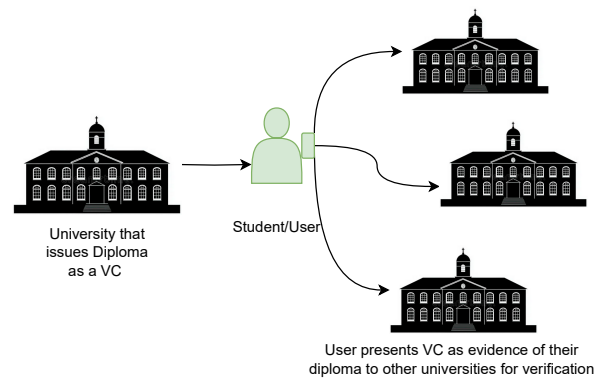


Figure 2. User-centric University Credentials

might also be setups where the issuer and verifier are the same entity, for instance if the credential in question is used to log in to only one service. In the SSO use-case, we could also have a setup where one entity issues credentials that are accepted for login purposes by multiple different services. It is clear that this setup requires that the three parties (the wallet, issuer, and verifier) have ways of communicating with each other.

There are two core security and privacy issues to take into account here. First, it should be impossible for anyone to convince a verifier they hold a credential when they don't. Second, the user should as far as possible be able to keep their personal data private for as long as they prefer.

For the first point, cryptographic tools, such as digital signature algorithms are used. Each credential is signed by the issuer, and this signature is what allows the verifier to determine that the credential indeed has been issued.

For the second, the decentralized approach goes a long way: the user only has to interact with the issuer once. There are functionalities that go further than this, for instance zero-knowledge proof based approaches, which allow the user to only reveal part of the information in the credential.

Various practical use cases exist for SSI in the academic literature, namely for blockchain-based identity and access management systems for IoT focusing on smart vehicles [3], addressing banking challenges in Know Your Customer (KYC) process [4], mapping SSI concepts to the healthcare domain [5], application of SSI in event ticketing systems [6], mobile SSI access control systems enabling secure peer-to-peer communication without internet connectivity [7], facilitating secure identity verification for decentralized energy systems [8], digital identity management [9] and facilitating border control access and travel data inspections [10].

For these use cases, SSI provides industry-specific benefits: in IoT, SSI can improve autonomy and resilience of the devices, whereas for the banking sector, SSI enables compliance adherence and efficiency of operations. For healthcare, an essential attribute is naturally privacy, for which SSI can be used to improve. In ticketing and related recreational applications, SSI can prevent fraud, and for access control,

SSI's benefit is to maintain resilience.

A. SSI as a user-centric technology

As described previously, SSI is at its core a more user-centric version of existing SSO solutions. This solution is more user-focused in three main ways: it allows the user flexibility, it prioritizes user control over their data, and it enables them to protect their privacy.

Due to the decentralized setup, the system provides flexibility to users. They can choose from a number of wallet applications, which may have different features, but are still able to receive and use credentials from any issuer.

SSI prioritizes user control over their personal data. Credentials are stored in the user wallet, and only shown to others when the user expressly consents to. On a technical level, this is achieved by using a *self-issued OpenID provider*. The user is able to make claims about themselves (possibly leveraging credentials they have previously received) without having to pass through an OpenID provider as is traditionally done in SSO setups.

Other functionalities can protect user privacy even further. Approaches based on zero-knowledge proofs can allow users to only reveal a subset of the information contained in their credentials, and wallet applications can keep track of the information revealed by users to ensure that not too much is revealed to any one party. Overall, the decentralized approach of multiple interoperable wallets allows wallet applications to cater to user needs both in privacy and for general user interface behaviour.

In terms of identity revocation in SSI, standards, such as [11] do not define specific revocation format or method. There exist several approaches to implement revocation in SSI, but a common system-wise challenge is that the mechanisms usually rely on some kind of centralization, which is in conflict with the basic principles of SSI [12]. Basically, implementations for revocation of identity are often based on the revocation lists similar with those familiar to Public Key Infrastructures (PKIs), often combining cryptographic accumulators with the mechanism. Another path is to build and utilize dependencies either between VCs, or between Issuer and Verifier, in which VCs validity can be limited for example by time or other parameter(s).

III. THE NEED FOR INTEROPERABILITY

As put forth by the original proposers of SSI, identities should be interoperable. This is important for a fundamental reason (identities as commonly understood are not tied to any one system), but is also necessary for a technical reason. If identities are user-controlled, they have to also be decentralized. A decentralized system collapses back to centralization if it is not possible for different systems to work together.

Due to the high level of effort necessary to receive documents containing sensitive personal data, user adoption is dependent on the systems to be easy to use after setup. If a single SSI wallet does not work across systems, using SSI becomes much more cumbersome.

Further, for wide adoption (which is necessary for a true digital identity) it is necessary for a large number of service providers to support authentication using SSI. If each provider has to implement the entire SSI stack for themselves, this will likely create a major obstacle. Therefore, it is important for new players to be able to rely on the existing infrastructure. This becomes much harder if the systems are not interoperable.

IV. CURRENT STANDARDS LANDSCAPE

Standards are a major enabler of interoperability. They provide a shared reference for providers to use. Fortunately, SSI already has well-established standards. Roughly, the standards can be broken down into four layers: technical, credentials, identity, and institutional. See Table I for a list of standards.

On the technical level, standards describe the way data is encoded. IETF has standardized JSON Web Tokens & Signatures (JWT & JWS) [13], which can be used to encode verifiable credentials. A different format, JSON-LD, has been standardized by W3C. The multicodec protocol, partly standardized by W3C, can be used to flexibly encode key material, which is necessary when defining DIDs.

These standards don't yet define the objects relevant to SSI. The necessary standards for SSI are defined by W3C, which standardize the core verifiable credentials data model [11], as well as various extensions of it. When combined with standardized encoding formats, it is then possible for parties to store, send and process verifiable credentials in a standardized way. In addition to defining the credential, these standards also define verifiable presentations, which is a format credentials are converted to in order to be presented to a verifier.

To be able to use these credentials, standards are still needed to describe how parties exchanging the credentials should operate. Here is where the OpenID Foundation standards [14] come into play. The protocols for both issuing and verifying credentials are heavily based on existing OpenID standards. The SSI-specific ones include OpenID for Verifiable Credential Issuance and OpenID for Verifiable Presentations. These define OpenID-based flows for a user to receive a credential (after authentication) and for authenticating using a verifiable presentation in a way interoperable with existing OpenID-flows.

The above flows leave some details open in the interest of future flexibility. This creates an issue for institutional players that would like to support the use of SSI. Consequently, various institutional players have stepped in to further specify interoperability conditions in given context. In the EU, EBSI [15] has defined a set of conformance tests for software, which specify the interaction between the components, and the EU digital identity wallet places requirements for member states in the EU [16]. In the private sector, organizations like the International Data Spaces Association [17] and GAIA-X [18] define SSI use together with data spaces.

V. OBSTACLES TO INTEGRATION

As laid out in Section IV, interoperability is necessary for self-sovereign identity to become a reality. This requires

TABLE I. STANDARDS AND ENTITIES RELEVANT FOR SSI

Standard	Layer	Role	Standards body or Entity
JOSE	Technical	Defines basic data objects	IETF
JSON-LD	Technical	Format for data objects	W3C
multicodec	Technical	Encodes key data	W3C
Verifiable Credentials Data Model	Credentials	Defines basic credential and presentation data formats	W3C
OIDC	Identity		OpenID Foundation
EBSI	Institutional	European Blockchain Services Infrastructure	European Commission
DID	Identity		W3C
EU digital identity wallet	Institutional		European Commission
IDSA	Institutional	Standards organization	International Data Spaces Association
GAIA-X	Institutional		GAIA-X Association

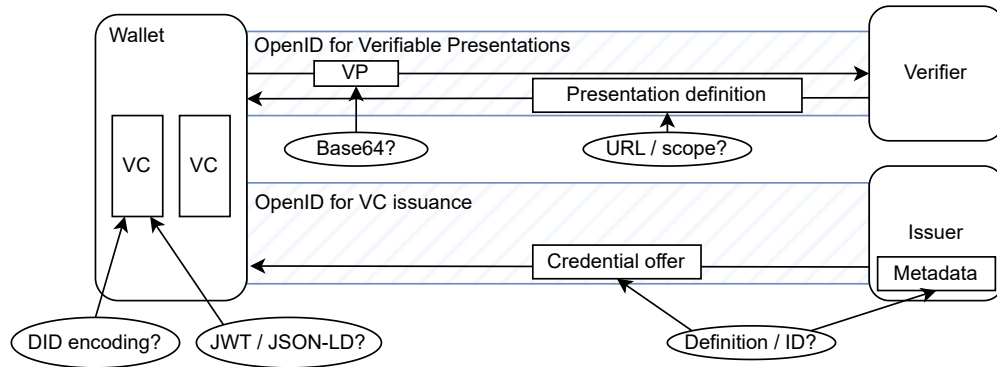


Figure 3. Diagram of problematic ambiguities in SSI flows. Each oval label identifies an area of divergence across implementations. See Section V for detailed explanations.

integration between various components. However, there are multiple obstacles to this integration work.

In this section, we point to places where the friction is especially high. Many of these have to do with incompatible formats. These issues were identified in the context of the Horizon Europe project TANGO [19], where the end goal was to use SSI for authentication in a data spaces settings. See Figure 3 for an overview.

The major issues caused by these obstacles are:

- 1) Friction when integrating systems, slowing down development work.
- 2) Technical impossibility, given time constraints, of making two systems interoperable.
- 3) Making projects harder to maintain due to needing code purely for compatibility reasons.

Overall, these issues reduce innovation by slowing down development work, and may lead to the establishment of non-interoperable ecosystems. For instance, friction caused by unclear formats might cause significant resources to be spent troubleshooting the issue.

A. VC format

The basic data structure of a verifiable credential is defined by the W3C. However, this standard is agnostic as to the

encoding of a credential. Two major competing encodings are JWT and JSON-LD. Both are JSON-based, but provide very different mechanisms for proving the authenticity of the credential. A software component that supports credentials in one format will not work for the other. As some functionalities, especially ones using advanced authentication technologies, are only available in one of the formats, they might fail to be supported in some systems. This increases the amount of work required for a software provider if they are to be interoperable with all systems. If two different providers use different formats, the user might well miss out on functionality important to them, for instance the use of zero-knowledge-proof enabled credentials. Further, they may need different versions of the *same* credential in different contexts, making the system cumbersome and less intuitive.

B. DID key encodings

With some DID methods, especially `did:key`, there is a need to encode a public key. This public key is to be read by the verifier when assessing whether the credential is valid. However, there are multiple different encoding formats for the key, even when the underlying algorithm is the same. This again forces providers to support each type, or risk hard-to-debug errors.

C. Presentation format

In order to use a verifiable credential, it needs to be converted to a verifiable presentation. In the `vp_token` flow, this verifiable presentation is then sent to the verifier for checking. However, the exact format this presentation takes can differ. It can be sent as a JWT, or that JWT can be encoded as a base64 string (which further can differ, e.g., whether it is URL-safe, and how padding is handled).

D. Breaking metadata changes

During the issuance flow, the credential issuer needs to expose some metadata itself, for instance to advertise the types of credentials it offers. However, there have been changes to this mechanism in the drafts of the relevant standard. In older versions, this information was contained in the credential offer itself, but later versions opt for an approach where the credentials offered are defined in the metadata of the issuer, and the credential offer identifies the credential being offered via an ID. This creates challenges especially in wallets that follow a frontend-backend architecture. In the first approach, the credential offer contains everything needed to describe the credential to the user. In the second, fetching of the information becomes more complicated, since the metadata of the issuer has to be specifically accessed.

E. Scope vs presentation definitions

When a verifier describes the type of credentials it is expecting, it has various different ways of doing this. A `presentation_definition` (possibly accessed through a `presentation_definition_uri`) describes the contents expected of the credential. However, the verifier can also make use of a `scope`, which is an arbitrary string that refers to some credential. However, this "scope" is hard for a wallet to be interoperable with, since the vocabulary is not defined in standards. To support a verifier that describes credentials using a `scope`, it is necessary to first define the meanings of the scopes between the organizations providing the wallet and verifier, and second for the wallet to specifically implement these scopes to properly support them.

VI. SOLUTIONS: WHAT WOULD HELP

The previous section outlines various integration obstacles observed when attempting to implement an SSI system in practice. This section contains proposals for changes in the landscape that would make integration work more seamless, and make it more likely for SSI to be used widely.

There are two main actors can help integration work: standardization bodies and technical specifications. The solutions suggested in this section apply to both, as well as the *drafts* of standards that currently are being used to build interoperable SSI systems.

Standard and draft versions should be included in machine-readable metadata of applications implementing SSI.

A major hurdle is uncertainty over the draft version of standard implemented by any given application. This can break interoperability either since there are breaking changes, or because a newer draft includes a new functionality. The draft version being clearly communicated in machine-readable metadata would make things smoother. Software could robustly pick the draft version to use, or at least give a clear indication of the issue (unsupported draft version). This would make it much easier for actors to develop systems, and stress-test the standard drafts themselves. This is especially relevant in the draft phase, where changes happen frequently.

Standards and technical specifications should together include exact technical details necessary for interoperability.

Standards support integration work as an intermediary between organizations by providing a common flow for them to agree on. Organizations can ideally build interoperable systems with little or no back-and-forth communication. However, this benefit is greatly reduced if the exact technical details are left out. This can cause issues that are hard to troubleshoot. This can to some extent be covered by standards themselves, but part of the work is left to technical specifications.

Standard drafts and technical specifications should clearly state which details are settled and which are not.

It is not always feasible or desirable to define all technical details. The standard might only be in the drafting phase, where work put into deciding and specifying technical details might be wasted if the final approach ends up being different. The authors of the standard might also want to avoid over-specifying the technical solution. This can leave space for innovation, support more use cases, or leave flexibility in case of further changes. For instance, setting in stone specific cryptographic signature algorithms would cause major issues if there were to be a need to change these (as in the case of the transition to post-quantum cryptography), and as such it is common practice to define these outside the standard itself. In any case, such areas where technical details are not (yet) agreed on, this should be made as clear as possible. This way organizations wishing to build interoperable systems have a clear basis for discussions. Further, it would be useful if there were a mechanism for implementations to communicate the specific choices made. Software can then adapt to the specific choices within the implementation.

A standard draft should specify any significant point which differs from those in a previous version.

This would help application developers update their systems as required. Version history is often provided in standard

drafts, but it would be especially helpful if these changes were communicated in machine-readable metadata.

Requirements related to user privacy should be set clearly by either standards or technical specifications.

SSI enables the use of many privacy-preserving technologies that can help users keep their data private. However, these technologies may require, for instance, the use of a specific format to store a credential, which in turn might not be supported by all wallets, issuers, or verifiers. This puts the user in a difficult position: they have adopted a technology, which promises them privacy, but in practice these functionalities might not be available. If full convergence of the standards is not feasible, actors wanting to support the takeup of SSI should place clear requirements for protection of user privacy, so that users can trust that the promise of user privacy is actually fulfilled.

VII. CONCLUSION AND FURTHER WORK

SSI is a promising identity technology, which can offer improvements to user experience and privacy. Due to the nature of SSI as a decentralized technology, interoperability is of great importance. Even though there is a robust set of standards for the different aspects of SSI, there are caveats that cause significant friction when integrating SSI systems in practice. We believe there are a number of approaches that could be taken now to make these systems easier to integrate, and consequently to foster further innovation in the space.

This paper is based on experiences gathered during our participation in the Horizon Europe TANGO project, with one set of software components. Determining if these issues are widespread, and identifying the obstacles faced in other projects could provide valuable input for standardization bodies. Furthermore, collecting feedback from a diverse set of stakeholders, such as wallet developers, end users, or institutional issuers would also be beneficial to the development of the SSI ecosystem. All of their differing perspectives could validate and contextualize the identified issues.

In future work, a cost-benefit analysis of the value brought by flexibility vs the cost imposed on integration efforts might allow for stronger recommendations for clarity in standards than the ones presented here, which defer to the value of flexibility.

ACKNOWLEDGMENT

This report has been funded by the TANGO project, which has received funding from the European Union's HE research and innovation programme under the grant agreement No. 101070052.

REFERENCES

- [1] C. Allen, "The path to self-sovereign identity," *Life with Alacrity*, 2016.
- [2] A. Tobin and D. Reed, "The inevitable rise of self-sovereign identity," *The Sovrin Foundation*, vol. 29, no. 2016, p. 18, 2016.
- [3] M. Naghmouchi, H. Kaffel, and M. Laurent, "An automatized identity and access management system for iot combining self-sovereign identity and smart contracts," *arXiv preprint arXiv:2201.00231*, 2022.
- [4] V. Schlatt, J. Sedlmeir, S. Feulner, and N. Urbach, "Designing a framework for digital kyc processes built on blockchain-based self-sovereign identity," *arXiv preprint arXiv:2112.01237*, 2021.
- [5] A. Siqueira, A. F. da Conceição, and V. Rocha, "User-centric health data using self-sovereign identities," *arXiv preprint arXiv:2107.13986*, 2021.
- [6] S. Ehrlich, V. Schlatt, and N. Urbach, "Exploring the use of self-sovereign identity for event ticketing systems," *Electronic Markets*, vol. 32, no. 4, pp. 1321–1339, 2022.
- [7] A. Enge, A. Satybaldy, and M. Nowostawski, "An offline mobile access control system based on self-sovereign identity standards," *Computer Networks*, vol. 219, p. 109434, 2022, ISSN: 1389-1286. DOI: <https://doi.org/10.1016/j.comnet.2022.109434>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1389128622004686> (Retrieved: 8/2025).
- [8] V. Gramlich, M.-F. Körner, A. Michaelis, and J. Strüker, "Ssi in the energy sector: A study," *Fraunhofer Institute of Applied Information Technology FIT*, 2023.
- [9] R. N. Zaeem *et al.*, "Blockchain-based self-sovereign identity: Survey, requirements, use-cases, and comparative study," *IEEE/WIC/ACM International Conference on Web Intelligence*, 2021. DOI: 10.1145/3486622.3493917.
- [10] P. Koskela, A. Karinsalo, J. Paananen, and L. Salmela, "Streamline border control with blockchain towards self-sovereign identity," in *3rd International Conference on Software Engineering, Security and Blockchain (SESBC 2022)*, Dubai, United Arab Emirates: AIRCC Publishing Corporation, 2022.
- [11] M. Sporny, G. Cohen, T. T. Jr, M. Jones, and I. Herman, "Verifiable credentials data model v2.0," W3C, Candidate Recommendation, Sep. 2024. [Online]. Available: <https://www.w3.org/TR/vc-data-model-2.0/> (Retrieved: 8/2025).
- [12] R. Chotkan, J. Decouchant, and J. Pouwelse, "Distributed attestation revocation in self-sovereign identity," in *2022 IEEE 47th Conference on Local Computer Networks (LCN)*, IEEE, 2022, pp. 414–421.
- [13] M. B. Jones, J. Bradley, and N. Sakimura, *JSON Web Token (JWT)*, RFC 7519, May 2015. DOI: 10.17487/RFC7519. [Online]. Available: <https://www.rfc-editor.org/info/rfc7519> (Retrieved: 8/2025).
- [14] K. N. Chadwick and J. Vercammen, "Openid for verifiable credentials," *OpenID Foundation*, 2022.
- [15] E. Commission, "European blockchain services infrastructure (ebsi) vc framework," European Commission, Regulation, 2024. [Online]. Available: <https://ec.europa.eu/digital-building-blocks/sites/display/EBSI/EBSI+Verifiable+Credentails> (Retrieved: 8/2025).
- [16] E. Commission, "European digital identity regulation," European Commission, Regulation, 2024. [Online]. Available: <https://digital-strategy.ec.europa.eu/en/policies/eudi-regulation> (Retrieved: 8/2025).
- [17] I. D. S. Association, "International data spaces association," IDSA, Webpage, 2024. [Online]. Available: <https://internationaldataspaces.org/> (Retrieved: 8/2025).
- [18] GAIA-X, Webpage. [Online]. Available: <https://gaia-x.eu/> (Retrieved: 8/2025).
- [19] TANGO, Webpage. [Online]. Available: <https://tango-project.eu/> (Retrieved: 8/2025).