A new Privacy Preserving and Scalable Revocation Method for Self Sovereign Identity - The Perfect Revocation Method does not exist yet

Andreas Freitag, University of Vienna

DRAFT: November 2022

Abstract:
Digital Identities are playing an essential role in our digital lives. Today, most Digital Identities are based on central architectures. Central Digital Identity providers control and know our data and thereby our Identity. Self Sovereign Identities are based on decentralized data storage and data exchange architecture, where the user is in sole control of his data and identity. Most of the issued credentials need the possibility of revocation. For a centrally managed Digital Identity system, revocation is not a problem. In decentral architectures, revocation is more challenging. Revocation can be done with different methods e.g. list based, cryptographic accumulators and with credential updates. A revocation method must be privacy preserving and must scale. This paper gives an overview of the available revocation methods, including a survey to define requirements, assess revocation groups against the requirements, highlights shortcomings of the methods and introduces a new revocation method called Linked Validity Verifiable Credentials.

Keywords:
Digital Identity, Decentralized Identity, Self Sovereign Identity, Privacy preserving, Accumulators, Revocation, Linked Validity Verifiable Credential

1 Introduction

This paper makes an assessment of existing revocation methods for Verifiable Credentials (VC) and a newly suggested revocation method, called Linked Validity Verifiable Credential (LVVC). The assessment covers privacy, scalability and maturity aspects for the use in Self Sovereign Identity (SSI) systems.
Existing revocation methods are organized in groups. The concept of LVVC is described. In a survey, requirements for privacy and technical performance for revocation have been derived. Based on these requirements, the assessment has been done.

1.1 Self Sovereign Identity

Digital Identities (DI) are playing an increasingly important role. In the 1960s, with databases, DI emerged. In this day's DI have been simple access lists. In the late 1990s, Microsoft introduced the first shared DI called Microsoft Passport. It was the predecessor of today Microsoft account. The goal was to have a single DI accessing different websites for e-commerce. All common DI have a significant disadvantage compared with physical identity documents. DI systems rely on central entities that control the system and, therefore, the identity of the user. This leads to privacy, control and access issues. The shortcomings of central DI led to the concept of SSI. The term SSI was minted and explained first by Christopher Allen in 2016 [1] together with a history of DI and his 10 principles of SSI. Documents that prove identity (e.g. passport) or belong to an identity (e.g. university certificates) and can be verified from a Verifier are called VC. One important SSI principle is the sole control over the DI from the Holder [1] [2]. This includes all issued VCs. To be in control, the Holder must store the VCs on his device.

1.2 Roles in an SSI system

In an SSI system are three main roles, the Issuer, the Holder and the Verifier [3]. The Issuer is issuing a VC and can perform a revocation of a VC. The Holder is in possession of a VC and can present a VC and a proof of non-revocation to the Verifier. The Verifier asks for the information and verifies it. The roles and the interaction between them is called "the triangle of trust". In any DI system, a trust layer is needed. A trust layer can be a Public Key Infrastructure (PKI) or a Decentralized Public Key Infrastructure (DPKI). A PKI or DPKI provides data which is needed for verification, e.g. the public key from the Issuer.

1.3 Revocation of an SSI Verifiable Credential

For many VCs, revocation is essential. The most common example is the driving licence. The simplest way to implement revocation is a central allow- or blocklists maintained by the Issuer. But this method contradicts the principles of SSI. The Verifier contacts the Issuer each time he verifies a VC. The Issuer knows when the credential is used and knows the Verifier. Privacy preserving and scalable revocation methods in an SSI solution must be developed, otherwise SSI systems cannot be implemented or compromises in terms of privacy would have to be made.

1.4 Delimitation

This paper focuses on the revocation of VCs in SSI systems from natural and legal persons. Nevertheless, revocation is also important in other areas as the Internet of Things (IoT). Privacy- and technical requirements differ in other areas.
Revocation methods are not combined with other cryptographic protocols like BBS group signatures [4] or primitives like Zero-Knowledge Proofs (ZKPs) even though they are necessary to create a full privacy preserving proof in an SSI system.

2 Contribution

The paper makes the following contributions:

- It provides an overview over revocation methods and a classification in different groups
- A new revocation method called Linked Validity Verifiable Credentials (LVVC) is defined and described
- Privacy- and technical requirements for revocation for VCs in an SSI system based on a survey within the SSI community are defined
- An assessment framework for revocation methods is defined
- An assessment of the revocation groups to determine the applicability is provided
- The shortcoming and open areas are highlighted

3 Prior Work and Grouping

This chapter gives an overview of revocation methods and organizes them in groups for the assessment.

3.1 List Based

List based revocation methods are the simplest method. Different implementations differ in their properties.

List Based: List based revocation methods are simple allow- or blocklists. The lists are publicly available or can be queried via an interface without restrictions. Examples are Certification Revocation Lists (CRL) in the X.509 certificate standard used for TSL [5].
List Based Hidden: In hidden list based revocation methods, the allow- or blocklist is hidden. A trusted party is necessary to manage the list and the access. Group signatures can be used. A group manager controls the list and ensures anonymity[6].
Compressed List: Compressed lists compress the information. The advantage is that the size of the list is smaller, and the storage, download, and query are efficient. Bloom filters[7, 8] or bit-arrays can be used to implement compressed list methods.

3.2 Cryptographic Accumulators

An accumulator is a one-way function that sums a large set of items into a single accumulator value. The membership of an included item can be proofed using the accumulator, the item itself, and a witness file [9, 10].

Asymmetric and Symmetric accumulators: Asymmetric accumulators need additional information, called a witness, for verification. Symmetric accumulators work without witness information. Examples for asymmetric accumulator are RSA based [9, 11, 12, 13, 14, 8], Elliptic Curve/Bilinear Pairing based [15, 16, 17] and Merkle tree based [18, 19, 20] accumulators. An example for a symmetric accumulator is a Bloom Filter.

Update properties: Cryptographic accumulators have different update properties. An update is an addition or/and deletion of an item. The definition is based on [21]. The static accumulator cannot be updated. Therefore, a static accumulator is not suitable for revocation. An additive accumulator can only include additional items. Items that are included can never be removed from the accumulator. A subtractive accumulator can only exclude items. A dynamic accumulator is additive and subtractive. Items can be included and excluded. A dynamic accumulator allows excluding and after that including the same element again.

Proofs: Cryptographic accumulators have different proofing properties. The definition is based on [21]. A positive accumulator can prove the membership of a certain item in an accumulator called membership proof. A negative accumulator can prove that an item is not included in an accumulator, called non-membership proof. A universal accumulator supports both membership and non-membership proofs. It must be possible for a verifier to verify the proof.

3.3 Credential Update

The validity of a credential is time-limited. If the Issuer chooses a short period of validity, e.g. 24h. The Holder must update his credential every 24h to keep the validity. The Issuer issues the credential again with a new issuance date [22].

4 A new Revocation Method: Linked Validity Verifiable Credentials

Linked Validity Verifiable Credentials (LVVC) are introduced in this paper as a new revocation method. A LVVC is a further development based on the principles of the credential update. A LVVC is a VC linked to another VC. The LVVC includes minimal information about the issuer and time of issuance. The time of issuance is important for the determination of the validity. The Holder must update the LVVC regularly, depending on the requirements of the Verifier. The advantage compared to a re-issuance of the linked VC without an additional LVVC [22] is the fixed size of the LVVC. A VC can vary in size and this can affect the scaleability. Another advantage of the decoupling is additional security and privacy. The update service from the Issuer does not require all the information about the issued VC. Only the VC identifier, Holder identifier, Issuer signing key and the revocation status are needed to issue a new LVVC.

In Figure 1 the process flows of issuance, revocation and verification are described. The figure includes a layer representing a PKI or a DPKI. The PKI or DPKI acts as a trust layer where the Verifier can obtain information about the Issuer and type of VC to be able to verify a VC and the LVVC.

Issuance Step 1: an additional LVVC, with the VC is issued to the Holder.

Revocation/Update Step 2,3: Revocation is performed indirectly with the LVVC. The LVVC includes an issuance date. If the Holder needs a current date, he must to request a new LVVC. If the VC is not revoked, the Issuer issues a new LVVC with the current issuance date.

Verification Step 4,5,6: The verifier initiates a verification request. The Holder creates a proof with the VC and LVVC. The Verifier performs the following verifications: Are the proofs validly signed? Are VC and LVVC linked? Does the issuance date of the LVVC meet these conditions?

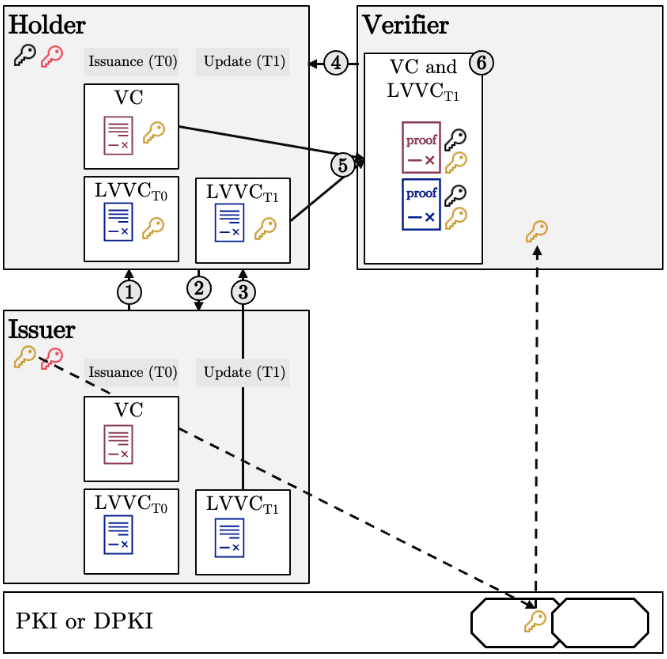

Figure 1: Process Flow LVVC

5 Survey

A survey to define the baseline and minimal requirements for the following assessment was carried out. The survey consisted of two parts. The first addresses privacy (correlation, linkage, knowledge of transaction data) and the second part technical requirements (storage space, computational effort, network). The survey was conducted with members of the SSI practice, who are working on SSI implementations, are part of an SSI consortium and/or are involved in standardization activities. An explanatory group was invited to the survey. The survey was not anonymous to ensure the qualification of the participants. To be considered in the evaluation, the participates had to provide an email address, name and company/consortium. The information was checked, and if the participants could not be verified, the answers were not considered in the evaluation.

5.1 Structure

A semi-structured approach is applied. The composition of the questions consists of thirteen closed and mandatory multiple-choice questions and eight open text fields where participants could add more information.

Privacy: The importance of data correlation, data linkage and traceability and the avoidance of collecting transaction data was queried. The participants were asked if a violation of a privacy aspect from a revocation method would be a reason not to use a revocation method.

Technical: The maximum acceptable storage space for the three different roles in the revocation process, the Issuer, the Holder and the Verifier was asked because each role use different hardware. An Issuer will run the service most probably on a server, the Holder will manage the credentials on a smartphone, and verification will be performed on a server or smartphone. Therefore, it is important to distinguish the requirements between the roles. As well as the maximum acceptable storage space, the acceptable computational effort depending on the role was queried too. Time is more crucial during a verification process than for an update process in the backend of the Issuer. Also, the acceptable network bandwidth for an Issuer was included in the survey. Some revocation methods require the transmission of additional information to each Holder. This information is often called witness. If the Issuer manages several millions of VCs in one accumulator, the amount of data can be significant, and therefore, the network bandwidth must be considered.

5.2 Results

23 participants answered the survey.

Privacy: The evaluation of the survey shows a uniform result in the privacy part. The answers in figure 2 show that for 74% - 83% privacy is important or very important.
Figure 3 shows that 68% would not use a revocation method that violate the three asked privacy aspects.

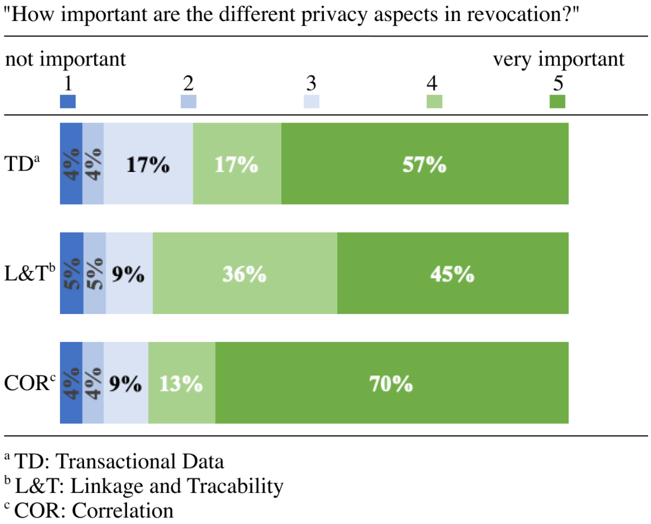

Figure 2: Privacy

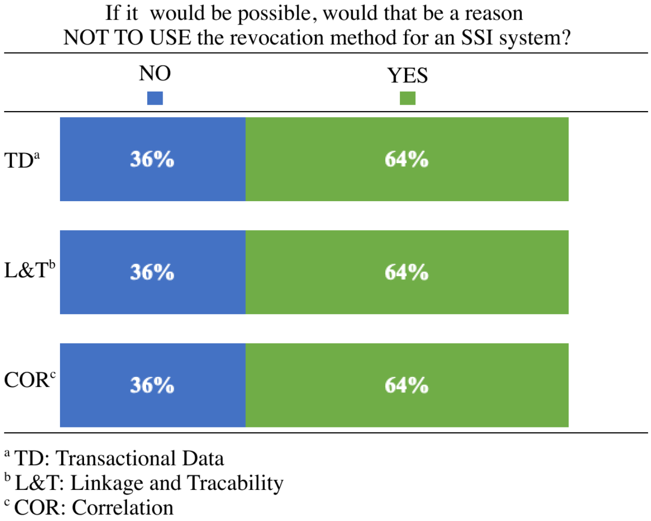

Figure 3: Privacy KO criteria

Technical: The evaluation of the questions (shown in figure 4, 5, 6, 7, 8 and 9) about the technical requirements does not show a consistent result. The answers are spread across the spectrum, with a high percentage of ”Don’t know” responses. The green bar represents the range where the median of all given answers is located.

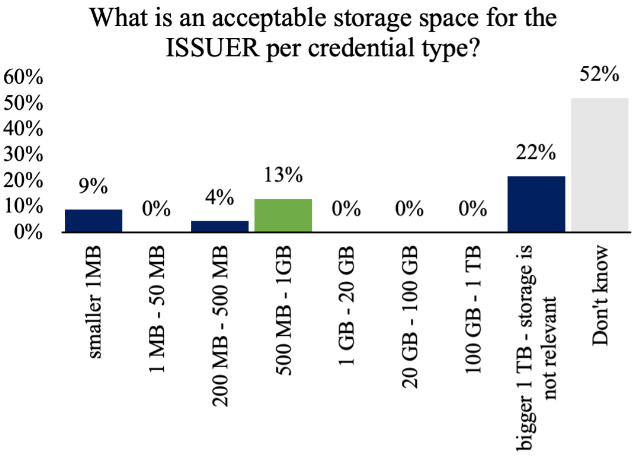

Figure 4: Requirements Storage Space Issuer

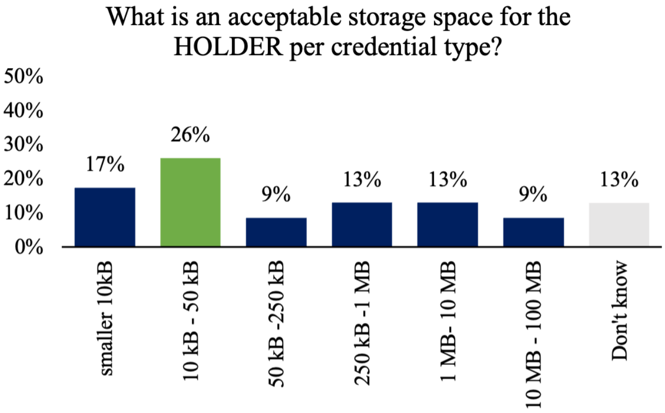

Figure 5: Requirements Storage Space Holder

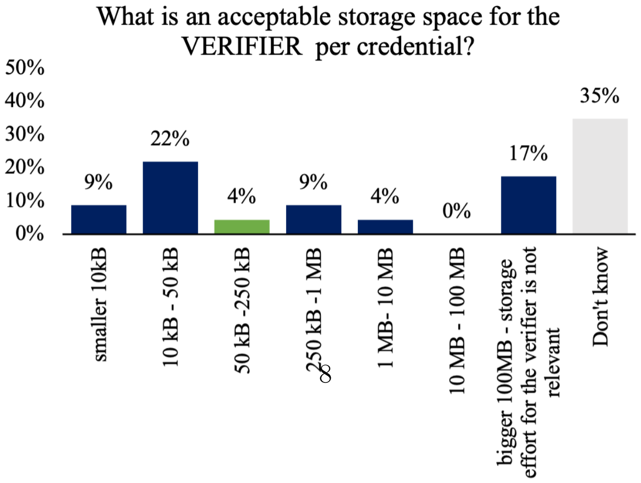

Figure 6: Requirements Storage Space Verifier

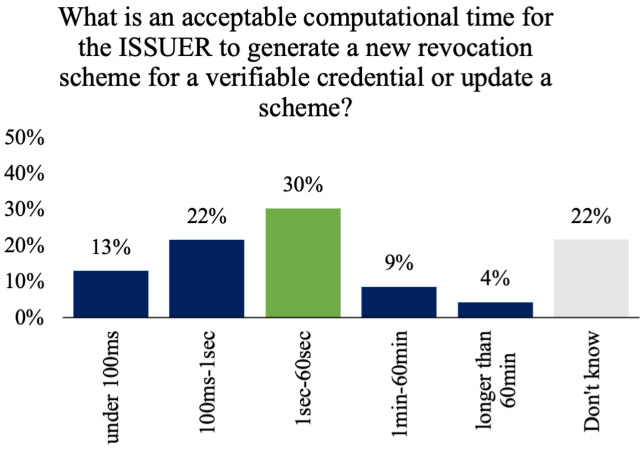

Figure 7: Requirements Computational Effort Issuer

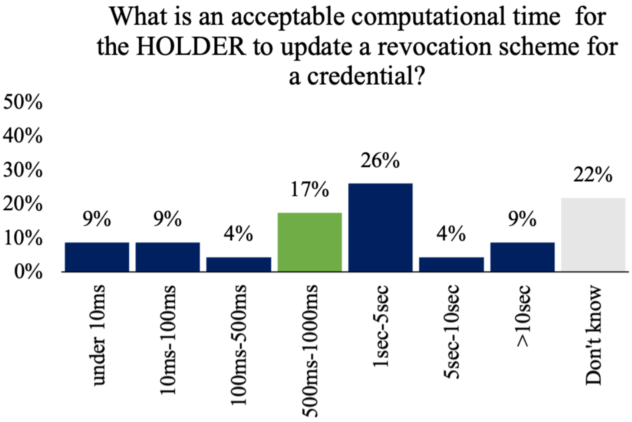

Figure 8: Requirements Computational Effort Holder

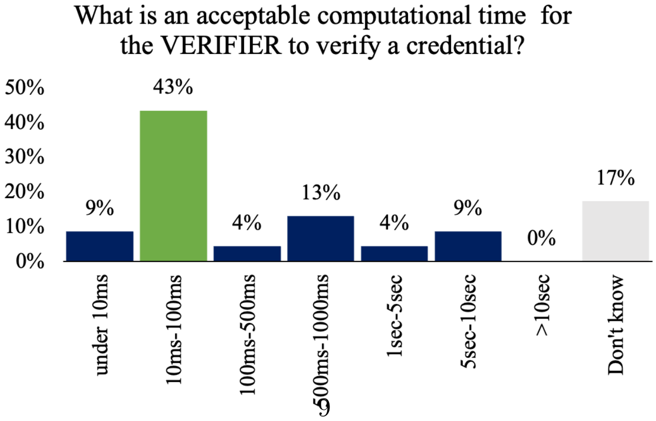

Figure 9: Requirements Computational Effort Verifier

6 Requirements for the Assessment

The assessment of revocation methods is based on privacy, technical, scalability and maturity aspects.

6.1 Privacy

Privacy for the Holder and a Verifier is a pre-condition of SSI. A revocation method must be privacy preserving to fulfill the principles of SSI. Therefore, the assessment has a stronger focus on privacy.

To prove that privacy is a key characteristic in SSI systems, the history of SSI and the characteristic must be considered. In the year 2005 Cameron [2] wrote a paper called "The Laws of Identity". The paper describes "laws" which must be fulfilled to raise the acceptance of DI systems. In a central system, the user must trust a central entity. Compliance cannot be controlled from the outside. With reference to Cameron [2] Christopher Allen wrote in 2016 a highly regarded blog post "The Path to Self-Sovereign Identity" [1] where he described 10 principles of SSI as a starting point for further discussion. Satybaldy, Nowostawski and Ellingsen [23] developed in the paper "Self-Sovereign Identity Systems: Evaluation Framework" an evaluation framework based on the work from Allen [1], Cameron [2] and added usability as an additional requirement. They defined eight overall characteristics for SSI systems. 2020 Naik and Jenkins [24] developed in the paper "Governing Principles of Self-Sovereign Identity Applied to Blockchain Enabled Privacy Preserving Identity Management Systems" a more comprehensive evaluation framework with 20 characteristics. Naik and Jenkins applied it to the Sovrin SSI framework. 2019 Ferdous, Chowdhury and Alassafi [25] introduced a formal model to describe and assess digital identity systems. In the literature describing SSI, privacy is a key topic.

The results from the survey in chapter 5 show that all three defined privacy aspects are relevant. If a revocation method violates one of the three defined privacy aspects, it should not be used for revocation.

6.2 Technical

The median from the survey results in chapter 5 is used as the baseline for the assessment. In table 1 a summary of the medians of all questions is shown.

Table 1: Summary Survey Results Median

	Issuer median	Holder median	Verifier median
Storage Space	500MB-1GB	10kB-50kB	50kB-250kB
Computational Time	1sec.-60sec.	500ms -1000ms	10ms-100ms

6.3 Scalability and Maturity

An important requirement is scalability. A revocation method in an SSI system needs to be able to cover millions of credentials to compete with existing central based identity systems. In this paper, scalability is defined as usable in

real world large-scale implementations with >1 million users and >1 million credentials. The results from the survey for computational effort and storage requirements in chapter 5 are used as a base for the assessment.
In the assessment, the maturity is rated on the basis of existing implementations. If the method is used in large-scale implementations, then it is rated high. If the method is in use but not large-scale and/or used cryptographic primitives are in a standardization process, it is rated medium mature. If the method is not in use and/or the used cryptographic primitives are at the research level and are not in a standardization process, it is rated as low

7 Definitions for the Assessment

In this chapter, interactions and processes, privacy levels and privacy aspects applied in the assessment are defined.

7.1 Interactions and Processes

Each revocation method requires interactions between Issuer, Holder, and Verifier. An interaction is defined as a contact between different roles. Interactions in the issuance-, revocation- and verification processes are assessed. The issuance process is the first issuance of a VC and additional steps necessary to enable revocation. In the revocation process, items will be revoked and data will be updated. In the verification process, the Verifier verifies the validity of a VC.

7.2 Privacy Aspects

The assessment included three privacy aspects. Privacy aspects are assessed from the Holder perspective.

- **Correlation:** If one party possesses information, that gives the possibility of linking the information with information from other parties. This can be anything that represents a unique identifier such as a specific HASH value, an identifiable credential or a public key that is used multiple times.
- **Transaction data:** transaction data is metadata about the usage of a VC.
- **Linkage:** A revocation has to be proofed at a certain point in time. The Verifier should be able to verify the validity of the VC exactly at this point. Linkage is defined as the possibility that the Verifier can check the validity of the VC in the past or in the future without the Holder's involvement.

7.3 Perspectives and Privacy Levels

In an SSI system, the Holder interacts with two different parties, the Verifier, and the Issuer. Therefore, the privacy level must be assessed from both perspectives. Privacy levels are influenced from the interactions described in chapter 7.1.

Privacy Level Holder to Issuer:
Revocation methods have different privacy levels in the relationship between Holder and Issuer.

- **Full Privacy:** The Issuer gets no information about the usage of the VC or/and the verifier.
- **Semi Privacy:** The Issuer gets information that the VC is used from the Holder or/and gets the information that a Verifier is performing a validation process on a VC issued by the Issuer.
- **No Privacy:** The Issuer knows the VC, the Holder and the Verifier in the validation process.

Privacy Level Holder to Verifier: Revocation methods have different privacy levels in the relationship between a Holder and a Verifier.

- **Full Privacy:** Full privacy is provided from anonymous revocation methods. Anonymous methods reveal only the current validity and nothing about the use or validity in the past and future.
- **Semi Privacy:** Semi privacy methods do not reveal everything to the Verifier or public. An additional piece of information is needed for verification and access.
- **No Privacy:** The validity of a credential can be verified by everybody, without restriction, and every time.

8 Assessment

The defined revocation groups in chapter 3 are assessed with the defined assessment criteria described in chapter 7.

8.1 Required interactions

The definition can be found in chapter 7.1. The analysis of the interactions is necessary for the determination of the privacy levels.

Table 2 shows, that list based revocation methods have an advantage in the issuance and revocation process. No interaction between any other roles is required because of revocation. But there are disadvantages in the verification process. In a list based approach, the Issuer or another third party needs to be contacted to validate the VC, this is called "calling-home".

In non-list based revocation methods, the Verifier does not contact the Issuer during the verification process. To make this possible, the Holder requires additional information from the Issuer, the witness. The witness is issued to the Holder during the initial issuance process and needs to be re-issued or updated after each revocation process. Cryptographic accumulator methods must update all witness information from all contained VCs. With the credential update or LVVC method, not all witnesses are affected. An update is performed only on VCs where the revocation status changed.

8.2 Holder Privacy

Based on the interactions defined in chapter 8.1, the privacy aspects from the Holder towards the Issuer and towards the Verifier are assessed. For the Holder to Verifier perspective, only correlation and linkage are relevant, as the Verifier is never involved when another Verifier verifies a VC. For the Holder to Issuer perspective, only the transaction data aspect is relevant. Linkage is irrelevant for the Issuer as he knows all the revocation information from his issued

Table 2: Required Interactions

	Issua.[a]			Revoc.[a]			Verif.[b]		
Group	I[d]	H[e]	V[f]	I[d]	H[e]	V[f]	I[d]	H[e]	V[f]
List Based	n	n	n	n	n	n	y	y	y
List Based Hidden	n	n	n	n	n	n	y	y	y
Compressed List	n	n	n	n	n	n	y	y	y
Cryptographic Accumulators	y	y	n	y	y	n	n	y	y
Credential Update	y	y	n	y	y	n	n	y	y
LVVC[g]	y	y	n	y	y	n	n	y	y

y: yes, the role performs an interaction with another role
n: no, the role is not involved in the process
[a] Issua.: Issuance Process
[b] Revoc.: Revocation Process
[c] Verif.: Verifier Process
[d] I: Issuer
[e] H: Holder
[f] V: Verifier
[g] LVVC: Linked Validity Verifiable Credential

VCs and if the Issuer has transaction data, he does not need to correlate the data. The assessment of the privacy aspects defines the privacy level.

Full privacy is only possible if correlation and linkage of data is impossible, and transaction data cannot is not collected. In the assessment, a correct implementation of the method is assumed. Every privacy preserving method can be implemented less privacy preserving. Cells marked with "y-n" reflect the dependency on a privacy preserving implementation, e.g. use of zero knowledge proofs and no-unique identifiers.

Holder to Issuer: All list based methods provide No Privacy. Every list based method require the Verifier to contact the Issuer and therefore, the Issuer knows when and from which Verifier the VC is used.
Cryptographic Accumulators, credential updates and LVVC provide full privacy as the Issuer learns nothing about the usage of the VC.

Holder to Verifier: The Privacy Level depends also on the design of the system. If the system uses unique identifiers or the generated proofs are always the same, it is possible for different Verifiers to correlate the data.
Linkage is possible with list based methods, but not with the non-list based revocation groups. The data provided for validation changed if the revocation status of one or more items changes. Therefore, the revocation status for an VC can only be checked at one point in time and not before this point or in the future.

Table 3: Holder Privacy

	Holder to Issuer		Holder to Verifier		
Group	TD[a]	PL[b]	COR[c]	LINK[d]	PL[a]
List Based	y	No Privacy	y	y	No Privacy
List Based Hidden	y	No Privacy	y-n	y-n	Semi Privacy
Compressed List	y	No Privacy	y	y	Semi Privacy
Cryptographic Accumulators	n	Full Privacy	y-n	n	Full Privacy
Credential Update	n	Full Privacy	y-n	n	Full Privacy
LVVC[e]	n	Full Privacy	y-n	n	Full Privacy

y: yes, the privacy aspects is violated
n: no, the privacy aspect is not violated
[a] TD: Transaction Data
[b] PL: Privacy Level
[c] COR: Correlation
[d] LINK: Linkage
[e] LVVC: Linked Validity Verifiable Credential

8.3 Scalability and Maturity

The basis for the assessment is defined in chapter6.3.

List Based Methods: List based methods and List Based Hidden methods are used in X 5.09 certificates with Certification Revocation List [5] or Online Certificate Status Protocol (OCSP)[5]. Therefore, the scalability and maturity are high.

Compressed List: The scalability and maturity of Compressed List methods depends on the implementation. If merkle trees or bit-arrays are used, the scalability and maturity are high, as these are used in large-scale production systems.

Cryptographic Accumulators: The only cryptographic accumulator method currently used for SSI systems is the RSA based accumulator in the Hyperledger Indy project [26]. A pilot implementation in the Province of British Columbia in Canada has revealed limitations regarding witness file size and computational effort[27]. 32.768 VCs included in the accumulator lead to a tail-file size of 8,4 MB and a proof generating time of 7 sec. with an iPhone12. The file must be downloaded from each Holder to calculate the witness. The file size grows linear with the number of VCs included. This method does not scale in a large-scale use case. Cryptographic Accumulators are not used in large-scale production environments and have limitations in scalability. Therefore, Cryptographic Accumulators are rated low regarding scalability and mid-low regarding maturity. The working group applied crypto in the Decentralized Identity Foundation (DIF) is working on new cryptographic accumulator methods [28] with the goal to improve scalability. The work is at the beginning and additional work in research and implementation needs to be done.

Credential Update and LVVC: For a VC, JavaScript Object Notation (JSON) [29] as the format standard and JSON Web Token[30] for signature can be used. They are used and established standards. Therefore, the maturity of revocation with a Credential Update and LVVC is rated high. The scalability depends on the size of the VC. In the Credential Update method, the size of the VC cannot be controlled. The Issuer can integrate unlimited attributes and data, therefore the scalability is rated high-medium. A LVVCs always has the same size. A JSON LVVC signed with BBS [4] is smaller than 1100 bytes. If an Issuer manages 1 million credentials, 1,1 Gigabyte of storage is needed. The payload transmitted to the Holder during an update is only 1100 bytes. Therefore, the scalability for the LVVC method is rated high.

Table 4: Scalability and Maturity

Group	Scalability	Maturity
List Based	high	high
List Based Hidden	high	high
Compressed List	high	high
Cryptographic Accumulators	low	medium-low
Credential Update	high-medium	medium
LVVC[a]	high	medium

[a] LLVC: Linked Validity Verifiable Credential

9 Conclusion

A perfect revocation method would be a method where interactions are reduced to a minimum. During issuance, no additional interactions should are necessary, as with list based revocation methods. During a revocation update, no interaction between the Issuer and the Holder should be necessary, as with list based methods and during verification there should be no interaction towards the Issuer, as with non-list based methods. At the time being, no revocation method combines all these interaction requirements.

The best method for privacy reserving revocation is a cryptographic accumulator combined with zero knowledge proofs. The assessment showed that cryptographic accumulators have limiters in scalability and maturity. More work needs to be done in research, implementation, and testing.

Therefore, the LVVC method is the most suitable for SSI implementations. LVVC offers better privacy than list based methods and can provide the needed scalability and maturity.

References

[1] Christopher Allen. *The Path to Self-Sovereign Identity.* 2016. URL: http://www.lifewithalacrity.com/2016/04/the-path-to-self-soverereign-identity.html (visited on 12/08/2020).

[2] Kim Cameron. *The Laws of Identity.* 2005. URL: www.identityblog.com.

[3] W3C. *Verifiable Credentials Data Model 1.0.* URL: https://www.w3.org/TR/vc-data-model/ (visited on 12/08/2020).

[4] Dan Boneh, Xavier Boyen, and Hovav Shacham. "Short Group Signatures". In: *Advances in Cryptology – CRYPTO 2004.* Ed. by Matt Franklin. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2004, pp. 41–55. ISBN: 978-3-540-28628-8. DOI: 10.1007/978-3-540-28628-8_3.

[5] *X.509 : Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks.* URL: https://www.itu.int/rec/T-REC-X.509/en (visited on 03/30/2021).

[6] David Chaum and Eugène van Heyst. "Group Signatures". In: *Advances in Cryptology — EUROCRYPT '91.* Ed. by Donald W. Davies. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 1991, pp. 257–265. ISBN: 978-3-540-46416-7. DOI: 10.1007/3-540-46416-6_22.

[7] Kaisa Nyberg. "Fast accumulated hashing". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).* Vol. 1039. ISSN: 16113349. Springer Verlag, 1996, pp. 83–87. ISBN: 3-540-60865-6. DOI: 10.1007/3-540-60865-6_45. URL: https://link.springer.com/chapter/10.1007/3-540-60865-6_45.

[8] Tolga Acar, Sherman S M Chow, and Lan Nguyen. *Accumulators and U-Prove Revocation.*

[9] Josh Benaloh and Michael de Mare. "One-way accumulators: A decentralized alternative to digital signatures". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).* Vol. 765 LNCS. ISSN: 16113349. Springer Verlag, 1994, pp. 274–285. ISBN: 978-3-540-57600-6. DOI: 10.1007/3-540-48285-7_24.

[10] Nelly Fazio and Antonio Nicolosi. *Cryptographic Accumulators: Definitions, Constructions and Applications.* 2002. URL: https://pdfs.semanticscholar.org/a611/cef6f0391bd5a8eec61b5cf0e1e1896a0dae.pdf.

[11] Niko BariĆ and Birgit Pfitzmann. "Collision-free accumulators and fail-stop signature schemes without trees". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).* Vol. 1233. ISSN: 16113349. Springer Verlag, 1997, pp. 480–494. ISBN: 3-540-62975-0. DOI: 10.1007/3-540-69053-0_33.

[12] Jan Camenisch and Anna Lysyanskaya. "Dynamic accumulators and application to efficient revocation of anonymous credentials". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).* Vol. 2442. ISSN: 16113349. Springer Verlag, 2002, pp. 61–76. ISBN: 3-540-44050-X. DOI: 10.1007/3-540-45708-9_5. URL: https://link.springer.com/chapter/10.1007/3-540-45708-9_5.

[13] Peishun Wang, Huaxiong Wang, and Josef Pieprzyk. "A New Dynamic Accumulator for Batch Updates". In: 2007, pp. 98–112. DOI: 10.1007/978-3-540-77048-0_8. URL: http://link.springer.com/10.1007/978-3-540-77048-0_8.

[14] Jiangtao Li, Ninghui Li, and Rui Xue. “Universal accumulators with efficient nonmembership proofs”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).* Vol. 4521 LNCS. ISSN: 03029743. 2007, pp. 253–269. ISBN: 978-3-540-72737-8. DOI: 10.1007/978-3-540-72738-5_17.

[15] Lan Nguyen. “Accumulators from bilinear pairings and applications”. In: *Lecture Notes in Computer Science.* Vol. 3376. ISSN: 03029743. Springer Verlag, 2005, pp. 275–292. DOI: 10.1007/978-3-540-30574-3_19.

[16] Jan Camenisch, Markulf Kohlweiss, and Claudio Soriente. “An accumulator based on bilinear maps and efficient revocation for anonymous credentials”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).* Vol. 5443. ISSN: 03029743. Springer, Berlin, Heidelberg, 2009, pp. 481–500. DOI: 10.1007/978-3-642-00468-1_27.

[17] Giuseppe Vitto and Alex Biryukov. *Dynamic Universal Accumulator with Batch Update over Bilinear Groups.* 777. 2020. URL: http://eprint.iacr.org/2020/777 (visited on 03/12/2021).

[18] Philippe Camacho et al. “Strong accumulators from collision-resistant hashing”. In: *International Journal of Information Security* 11.5 (2012), pp. 349–363. ISSN: 16155262. DOI: 10.1007/s10207-012-0169-2.

[19] Leonid Reyzin and Sophia Yakoubov. *Efficient Asynchronous Accumulators for Distributed PKI.* 718. 2015. URL: http://eprint.iacr.org/2015/718 (visited on 12/23/2020).

[20] Mahabir Prasad Jhanwar and Pratyush Ranjan Tiwari. *Trading Accumulation Size for Witness Size: A Merkle Tree Based Universal Accumulator Via Subset Differences.* 1186. 2019. URL: http://eprint.iacr.org/2019/1186 (visited on 12/14/2020).

[21] Foteini Baldimtsi et al. “Accumulators with Applications to Anonymity-Preserving Revocation”. In: *Proceedings - 2nd IEEE European Symposium on Security and Privacy, EuroS and P 2017.* 2017, pp. 301–315. ISBN: 978-1-5090-5761-0. DOI: 10.1109/EuroSP.2017.13. URL: https://ieeexplore.ieee.org/document/7961987/.

[22] Jan Camenisch, Markulf Kohlweiss, and Claudio Soriente. *Solving Revocation with Efficient Update of Anonymous Credentials.* URL: http://godot.be/eidgraphs..

[23] Abylay Satybaldy, Mariusz Nowostawski, and Jørgen Ellingsen. “Self-Sovereign Identity Systems: Evaluation Framework”. In: Mar. 6, 2020, pp. 447–461. ISBN: 978-3-030-42503-6. DOI: 10.1007/978-3-030-42504-3_28.

[24] N. Naik and P. Jenkins. “Governing Principles of Self-Sovereign Identity Applied to Blockchain Enabled Privacy Preserving Identity Management Systems”. In: *2020 IEEE International Symposium on Systems Engineering (ISSE).* 2020 IEEE International Symposium on Systems Engineering (ISSE). ISSN: 2687-8828. Oct. 2020, pp. 1–6. DOI: 10.1109/ISSE49799.2020.9272212.

[25] Md Sadek Ferdous, Farida Chowdhury, and Madini O. Alassafi. “In Search of Self-Sovereign Identity Leveraging Blockchain Technology”. In: *IEEE Access* 7 (July 2019). Publisher: Institute of Electrical and Electronics Engineers (IEEE), pp. 103059–103079. ISSN: 2169-3536. DOI: 10.1109/access.2019.2931173.

[26] *0011: Credential Revocation — Hyperledger Indy HIPE documentation.* URL: https://hyperledger-indy.readthedocs.io/projects/hipe/en/latest/text/0011-cred-revocation/README.html (visited on 11/18/2022).

[27] *Indy Tails Server and Metrics.* original-date: 2020-04-30T17:49:14Z. Nov. 4, 2022. URL: https://github.com/bcgov/indy-tails-server (visited on 11/18/2022).

[28] *DIF working group applied crypto - revocation.* original-date: 2021-09-07T13:01:14Z. Aug. 18, 2022. URL: https://github.com/decentralized-identity/revocation (visited on 11/18/2022).

[29] Tim Bray. *The JavaScript Object Notation (JSON) Data Interchange Format.* Request for Comments RFC 7159. Num Pages: 16. Internet Engineering Task Force, Mar. 2014. DOI: 10.17487/RFC7159. URL: https://datatracker.ietf.org/doc/rfc7159 (visited on 11/21/2022).

[30] Michael Jones, John Bradley, and Nat Sakimura. *JSON Web Token (JWT).* Request for Comments RFC 7519. Num Pages: 30. Internet Engineering Task Force, May 2015. DOI: 10.17487/RFC7519. URL: https://datatracker.ietf.org/doc/rfc7519 (visited on 11/21/2022).