KICS The Korean Institute of Communications and Information Sciences

Available online at www.sciencedirect.com

ScienceDirect

ICT Express xxx (xxxx) xxx

www.elsevier.com/locate/icte

Selective disclosure in digital credentials: A review

Šeila Bećirović Ramić[a,*], Ehlimana Cogo[a], Irfan Prazina[a], Emir Cogo[a], Muhamed Turkanović[b], Razija Turčinhodžić Mulahasanović[a], Saša Mrdović[a]

[a] *University of Sarajevo, Faculty of Electrical Engineering, Bosnia and Herzegovina*
[b] *University of Maribor, Faculty of Electrical Engineering and Computer Science, Slovenia*

Received 11 January 2024; received in revised form 22 May 2024; accepted 23 May 2024
Available online xxxx

Abstract

Digital credentials represent digital versions of physical credentials. They are the cornerstone of digital identity on the Internet. In order to enhance privacy, different authors implement selective disclosure in digital credentials, allowing users to disclose only the claims or attributes they want. This paper gives an overview of the most influential articles for selective disclosure, a chronology of the evolution of the methods, and a list of strategies and approaches to the problem. We identify the categories of approaches and their advantages and disadvantages. In addition, we recognize research gaps and open challenges and provide potential future directions.

Keywords: Selective disclosure; Anonymous credentials; Verifiable credentials; Zero-knowledge proof

Contents

* Corresponding author.
E-mail addresses: sbecirovic1@etf.unsa.ba (Š.B. Ramić), ekrupalija1@etf.unsa.ba (E. Cogo), iprazina1@etf.unsa.ba (I. Prazina), ec15261@etf.unsa.ba (E. Cogo), muhamed.turkanovic@um.si (M. Turkanović), rturcinhodzic@etf.unsa.ba (R.T. Mulahasanović), smrdovic@etf.unsa.ba (S. Mrdović).
Peer review under responsibility of The Korean Institute of Communications and Information Sciences (KICS).

https://doi.org/10.1016/j.icte.2024.05.011

1. Introduction

Identification documents, passports, driver's licences and medical IDs are objects people carry with them and are a part of their daily lives. Even though they contain digital proofs, they are still paper-based credentials. In the world with advanced digitalization, a digital equivalent of this "paper" is required. That digital equivalent is called a digital credential. Unfortunately, this term is still used confusingly in different fields of computer science, computer security and cryptography because it is still evolving. A simple password is sometimes considered a digital credential; other times, a signed certificate is a digital credential [1].

This paper focuses on digital credentials that are pieces of evidence of an individual's qualifications, claims or achievements, also called attestations. We especially focus on anonymous credentials (AC), attribute-based credentials (ABC) and the relatively new verifiable credential (VC) formats or types. VCs represent all the information that a physical credential represents, but with digital signatures that make them more tamper-evident and more trustworthy than their physical counterparts. VCs are not just a digitally signed physical credential representation but are also standardized and can be based on schemes or contexts [2].

Each version of "digital credentials" offers much greater security than the physical non-identity object. They should enable the holder to determine when, how, and to which extent they want to reveal their information. As such, one key characteristic is data minimization and selective disclosure of attributes [3]. Selective disclosure enables users to share only the information they want with their specific parties. With selective disclosure, access to data is limited, especially to personal data. This principle can be illustrated by using the following example (shown in Fig. 1): *An individual graduated from the university and got their degree as a digital credential that contains their name, student number, degree name, academic score, etc. When applying for a job, they can selectively disclose only their name and degree name instead of sharing their academic score or other items.*

Selective disclosure represents a mechanism for preserving privacy for individuals and organizations. It can be seen as a privacy design pattern that enhances privacy and security. When implemented, it can provide the following benefits for both privacy and security [4,5]:

- Data minimization — sharing the minimal amount of necessary information reduces the amount of data collected and, as such, decreases the risk of data breaches and privacy violations;
- Compliance with data regulations — allows organizations to comply with data protection regulations General Data Protection Regulation (GDPR) [6] and the California Consumer Privacy Act (CCPA) [7] by minimizing the amount of collected personal data;
- Enhancing trust with users — allowing the users control over their data and shared pieces of information builds trust;
- Access control — allowing users to choose to whom to disclose data and which resources they can use to access the shared data.

1.1. Aim and contribution

In recent years, selective disclosure has been the focus of many approaches doing foundational work on this subject, especially in VCs. Different techniques and algorithms are used to achieve selective disclosure of digital credentials. Some approaches combine cryptographic and zero-knowledge proof methods, and some use blockchain in digital credential systems because of the decentralization. Zero-knowledge proof or zero-knowledge protocol (ZKP) is a method by which one party can prove to the other party that a given statement is true while avoiding sharing any information beyond the mere fact of the statement's truth [8]. Blockchain is a public or private distributed ledger built on a peer-to-peer network. It enables agreements on transactional data and sharing across a network of untrusted participants without relying on a central trusted authority [9].

Because this is a fast-evolving field, this paper aims to provide insight into the algorithms used, summarize the existing approaches and identify possibilities for future work. It provides an overview of the most influential articles on the topic of selective disclosure, a chronology of the evolution of the methods, and a list of strategies and approaches to the problem. Overview of methods and credentials raises the following research questions that we aim to answer:

- RQ1: Which selective disclosure forms and types exist, and what methods are used to achieve it?
- RQ2: Are there differentiations between selective disclosure methods used depending on the digital credential definition or format?
- RQ3: Which methods use zero-knowledge proof?
- RQ4: Which methods are built on blockchain?

Answers to these questions will provide the following contributions:

- Answer to the first question shows all the proposed methods with their (dis)advantages, and we see the current trends for achieving selective disclosure. We present differentiation and categorization of different types and formats for selective disclosure while introducing a new one, ZKP;

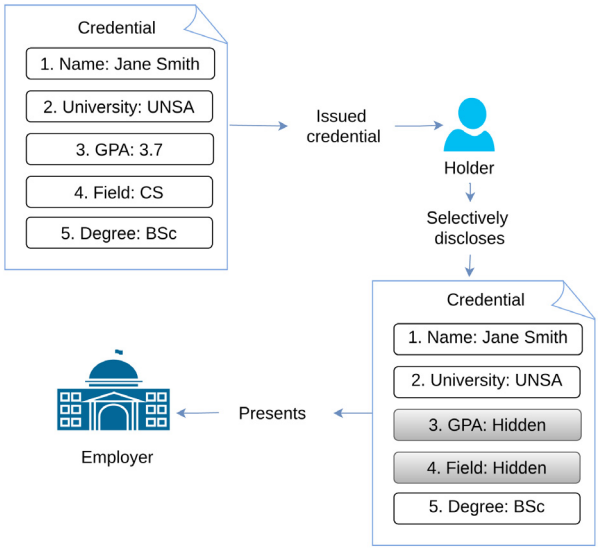

Fig. 1. Selective disclosure scenario.

- Answer to the second question shows how much influence credential definition or format has on a selective disclosure method, whether there are common themes and issues, and which methods are not used in newer formats. Finding common themes and issues can help researchers focus on them;
- Answer to the third question shows which role ZKP has in selective disclosure, how much it is used and if it is necessary;
- Answer to the fourth question shows the future trends in implementing credentials using blockchain.

This paper aims to provide an overview of the state-of-the-art selective disclosure within digital credentials. The overview presents the complex subject nature of selective disclosure in a structured and straightforward manner, which has not been done before in detail. As technology changes and evolves, especially in this field, it is necessary to know what is currently relevant and which issues remain unresolved.

1.2. List of abbreviations

Multiple abbreviations of names for technologies, terms, and approaches appear in this paper. To avoid the problem of confusion and make it easier to find the meaning of these abbreviations, we created a table of all the abbreviations that appear in this work. Moreover, explaining abbreviations in the text where they appear can sometimes disrupt the reading flow. In Table 1, we list all used abbreviations and their meanings.

2. Preliminaries

For a better understanding of the subject and the interconnection between the terms used, in this section, we will give a short historical overview of credentials and tools used for selective disclosure.

Table 1
List of abbreviations.

Abbreviation	Meaning
ABC	Attribute-based credential
AC	Anonymous credential
BBS	Boneh–Boyen–Shachum signature
BLS	Boneh–Lynn–Shacham signature
CCPA	California Consumer Privacy Act
CL	Camenisch–Lysyanskaya signature
DHE	Diffie–Hellman Exponent
DID	Digital identifier
DIHAC	Double issuer-hiding attribute-based credentials
EBSI	European Blockchain Services Infrastructure
EC	Erasure coding
ECC	Elliptic curve cryptography
ECDL	Elliptic curve discrete logarithm
eIDAS	Electronic Identification, Authentication, and Trust Services
ETSI	European Telecommunications Standards Institute
GDPR	General Data Protection Regulation
HM12	Hajny–Malina credential scheme
HMAC	Hashed message authentication codes
ICDH	Computational DiffieHellman
IPFS	Interplanetary file system
JSON	JavaScript Object Notation
MAC	Message authentication code
MDSA	Minimal Disclosure Signature Authentication
NFT	Non-fungible token
NIST	National Institute for Standards and Technology
NIZK	Non-interactive zero-knowledge proof
pABC	Privacy-enhancing attribute-based credential
PET	Privacy-enhancing technology
PKI	Public key infrastructure
PS-MS	Pointcheval–Sanders Multi-Signatures
RSA	Rivest–Shamir–Adleman encryption
SBT	Soul-bound tokens
SPK	Signature proof of knowledge
SPS	Structure-preserving signatures
SPS-EQ	Structure-preserving signatures on equivalence classes
SSI	Self-sovereign identity
URS	Unlinkable redactable signature
VC	Verifiable credential
XML	Extensible Markup Language
zk-SNARK	Zero-knowledge succinct non-interactive arguments of knowledge
ZKP	Zero-knowledge proof
ZKPK	Zero-knowledge proof of knowledge

The first significant contribution to digital credentials, AC systems, is connected to untraceable or anonymous payment improvements. David Chaum introduced blind signature protocols as a novel cryptographic primitive in 1983 [10] for untraceable or anonymous payments. Blind signatures can be publicly verified against the original, unblinded message in the manner of a regular digital signature. These blind signatures allowed Chaum to give the theoretical construction of AC systems in 1985 [11]. ACs allow users to prove possession of credentials or reveal information they want while maintaining anonymity [12].

It should be noted that Chaum's credentials are also known as a pseudonym system, where each user is known by their pseudonym to the organization and credentials are represented to the organization as pseudonyms that cannot be linked together. Theoretical development progressed further with papers by Ivan Bjerre Damgard [13], who focused on weak untraceability, and by Stefan Brands [14,15]. Brands introduced secret-key certificate schemes, creating Brands blind signatures that are a foundation of Microsoft U-Prove solution.

Camenish and Lysyanskaya created the first formalization of the AC system in paper [16]. Later, they published a series of papers [17–19], where they developed a signature with

efficient protocols known as CL signature. In their work, they defined and achieved the basic properties of ACs:

- Anonymity — Each user is anonymous in the system;
- Untraceability — It is impossible to track the user's usage of credentials;
- Unforgeability — It is not possible to forge a credential;
- Unlinkability — Usage of the same credential several times should not be linkable.

Besides basic properties, they defined and achieved additional desirable properties:

- Non-transferability — Users cannot share someone else's credentials;
- Selective disclosure — Users can choose the attributes they want to reveal;
- Revocation — It is possible to revoke a credential;
- Malicious user identification — Ability to recognize a malicious user.

Their scheme represents a building block for IBM's Identity Mixer (Idemix).

Dan Boneh, Ben Lynn and Hovav Schacham developed the BLS signature, a short group signature [20] built on bilinear pairing and elliptic curve. With C. Gentry, they proposed a solution that aggregates these signatures: Multiple signatures generated under multiple public keys for multiple messages can be aggregated into a single signature [21]. These signatures are extensively used in the Ethereum blockchain [22] and are the proposed solution for healthcare credentials [23].

Dan Boneh, Xavier Boyen, and Hovav Shacham continued the work on ACs, developing a short group signature known as BBS [24]. Their scheme is built on pairing-based elliptic curve cryptography (ECC). It was improved in paper [25] and is now referred to as a BBS+ signature scheme.

Further advancements were achieved with the real-life implementations of ACs, U-Prove and Idemix, where authors tried to improve or implement them for a specific research field. U-Prove [26] is based on blind cryptographic protocols designed by Stefan Brands, focusing on user-centric identity management where digital identity is connected to tamper-resistant devices such as smart cards. Brands founded Credentica in 2004 and developed U-Prove. U-Prove was acquired by Microsoft in 2008 [27].

In 2002, Jan Camenisch and Els Van Herreweghen from IBM presented Idemix, built for the PRIME/PRIMELIFE project, an AC system based on the CL signature scheme that allows anonymous, yet authenticated and accountable, transactions [28,29]. Both solutions implement selective disclosure using their underlying cryptographic primitives, which will be explained in the following sections.

The development of U-Prove and Idemix resulted in an EU-funded project known as Attribute-based Credentials for Trust (ABC4Trust) 2010–2015 [30]. The project aimed to define a typical unified architecture for federating and interchanging different privacy ABC systems. ABCs are defined as a form of authentication mechanism that allows one to flexibly and selectively authenticate different attributes about an entity without revealing additional information about the entity. Privacy ABCs (pABCs) allow holders to reveal and prove the minimal information required. Both Idemix and U-Prove were integrated into its architecture. Besides them, further schemes were introduced. One of them is HM12 scheme [31] with discrete logarithm commitments. The other one is open source IRMA app ("I Reveal My Attributes") known as the Yivi app [32], based on the Idemix ABC scheme which supports privacy-preserving features. Due to the popularity of Idemix, there are several papers discussing selective disclosure in specific credentials using CL signatures such as standard Java Cards [33], auditing [34], e-health [35], electronic coupons [36].

With advances in blockchain technology, the Linux Foundation founded Hyperledger as a collection of open-source blockchain projects. IBM cofounded Hyperledger Fabric (modular, permissioned blockchain framework) and imported Idemix into the project. In 2017, Everynym and Sovrin Foundation donated their project aimed at building a Self-sovereign identity (SSI) platform to Hyperledger, creating Hyperledger Indy. Cryptographic modules in Hyperledger Indy were then imported to Hyperledger Ursa, a project managing shared cryptographic libraries. The first implementation of AC used CL signatures, while the second was based on BBS+ signatures [37].

As the conversation on identity moved towards decentralized, user-centric identity, SSI was defined. SSI is an identity management model where each digital identity is controlled and managed by the entity to which the identity and related data belongs [38].

A VC is an open standard for digital credentials introduced in 2019 [2]. It represents a tamper-evident credential whose authorship is cryptographically verifiable. Since VCs are an open standard, more and more work is being done to enable anonymous credential privacy-preserving properties that enable data minimization, selective disclosure, and correlation resistance. Several solutions focused on anonymous VCs are implemented using BBS+ signatures, CL signatures, hash Merkle trees, selective disclosure — JSON web tokens (SD-JWTs), Hyperledger AnonCreds and others in recent years [37, 39,40]. Currently, the main focus is on BBS+ signatures, and as such, there are proposed applications of credentials for the healthcare sector [41], electronic voting [42]. With SSI and VCs, research on credentials, selective disclosure and revocation is more relevant each day.

One concept or requirement for an ACs remained throughout this entire previously explained development of digital credentials: Selective disclosure.

3. Related surveys

This section gives an overview of the papers that survey the problem of privacy, credentials and specific analysis methods.

In paper [43], Kaaniche et al. overview privacy-enhancing technologies (PETs). They identify a taxonomy and classify

the services into three groups. The paper reviews anonymous certification, namely anonymous ABCs, with comparative signature schemes (sanitizable signatures, attribute-based signatures, group signatures, blind signatures) analysis regarding security and functional requirements, i.e. unforgeability, anonymity, multi-show unlinkability, selective disclosure and traceability. They overview active Idemix and U-Prove systems.

An overview of existing technologies and requirements for vaccination certificate services is given in [44]. Corici et al. provide a compact survey to move towards interoperable certificates. They offer a summary of possible solutions and signatures for selective disclosure in VCs.

Mukta et al. in paper [45] give a survey on data minimization techniques in blockchain-based healthcare. One possibility for data minimization is selective disclosure, where they focus more on solutions presented solely for healthcare.

In paper [46] Flamini et al. give a first appraisal of cryptographic mechanisms for selective disclosure in VCs. They describe the structure of tools and compare them in terms of the performance and the size of the associated verifiable presentation (VP). Authors analyzed salted hash lists, Merkle hash trees, CL and improved BBS+ signatures.

To the authors' knowledge, all prior surveys focused on selective disclosure in a specific research area or on the performance of the most common methods used. There are no papers focused on identifying different methods used for different credential types.

4. Research methodology

The Systemic Literature Review defined by Kitchenham [47] was chosen as the primer research method. The chosen research methodology has three steps: searching, filtering and evaluating papers. In the first step, we included a set of relevant papers as extensive as possible. In the second step, we reduced the number of articles so that only those that deal directly with the topic and are of sufficient quality remained. In the last step, we performed a deeper analysis of the works to gain a better understanding. With this strategy, we optimized the time spent considering each paper. As a result, we were able to study the remaining articles in detail within a reasonable time.

4.1. Search methodology

We used different indexing services for finding scientific papers. All of the documents considered are written in the English language. They were peer-reviewed, except for articles written between 2018 and 2023, where we included preprints due to the recent faster developments in the area. Older preprints were excluded because of their questionable relevancy. We used the following search engines for obtaining the relevant literature: Science Direct, ACM Digital Library, IEEE Xplore, Scopus and Google Scholar. Unfortunately, the term "selective disclosure" is too broad and used in other research areas, so we combined it with "credential", "data minimization" and "data minimisation". The search term used was:

Table 2
Results by each search engine.

Search engine	Number of papers
Web of Science	14
IEEE Xplore	20
Scopus Preview	59
Science Direct	79
ACM Digital Library	146
Google Scholar	150
Total	**468**

Table 3
Inclusion and exclusion filters.

Inclusion	Exclusion
Peer reviewed	Not peer reviewed
English language	Chapters in books
Accessible and indexed	Short papers
Preprints from 2018 and onwards	Master or PhD theses

credential* AND ("selective disclosure" OR "data minimisation" OR "data minimization")

Results from Google Scholar were limited to the first 15 pages due to the very high number of resulting pages. After the preliminary search, more than 400 papers were found, with exact numbers per search engine given in Table 2.

4.2. Filtering methodology and criteria

All papers in the search were filtered using steps shown in Fig. 2. The first step was to remove duplicate articles. Afterwards, we screened papers by title and abstract, which discarded those that did not cover the topic of interest. Afterwards, we filtered the remaining papers by the rules of inclusion and exclusion formulated in Table 3. We included papers that are peer reviewed, accessible, indexed and in English for easier understanding. Preprints from the previous five years are also included because of their contribution, taking into consideration publication time. We excluded non-peer-reviewed papers, chapters in books that usually reference the papers used, short papers and Master's or PhD theses because they are generally published as papers. We finished filtering fast due to the analysis of abstracts and conclusions only, which does not require deep paper analysis.

Further filtering required extensive analysis of papers. Papers needed to follow the rules:

- The research paper's goal is similar to implementing selective disclosure features in credentials.
- There is a defined methodology and explanation for achieving selective disclosure.
- For the implementation, there should be no reliance on existing solutions for the issue.

With these filtering methods, only 30 papers remained. The detailed analysis and evaluation of the final corpus of relevant literature will be explained in the next section. This analysis will attempt to answer the research questions defined in Section 1.

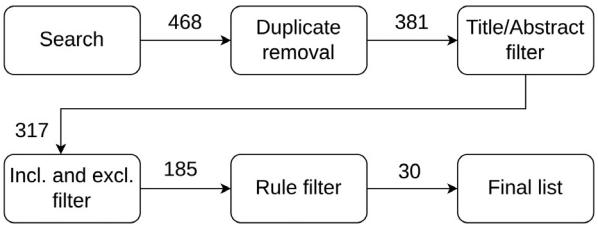

Fig. 2. Steps of research process and resulting number of papers.

5. Results

This section gives an analysis of the papers and answers the previously defined research questions.

5.1. RQ1: Which selective disclosure forms and types exist, and what methods are used to achieve it?

Authors of [48] define the following methods for selective disclosure:

- Atomic Credentials — Credentials which consist of a single claim. The atomicity allows the holder to disclose only claims which need to be revealed.
- Selective disclosure signatures — Certain signature schemes support selective disclosure natively.
- Hashed values — Credentials are issued containing all of the claims, but they represent hashed values.

As the simplest solution, atomic credentials can be found as a subset in both other methods and are unwieldy to manage due to the sheer number of credentials and higher communication overhead. There is also no guarantee that a proper pairing of two claims is possible, meaning that we can show credentials with two nonsensical or incorrect claims about a singular subject. Therefore, these methods should only be considered further for selective disclosure in ACs when the issues of pairing and size are solved.

Discussion of selective disclosure includes ZKP as well. Specific solutions allow for different ZKP protocols, while others exclusively implement selective disclosure relying on ZKP. Therefore, we identify one new category for selective disclosure, ZKP. We look at solutions that use hash-based approaches, selective disclosure signature approaches, ZKPs, and those that combine two or three to build a method for selective disclosure. An overview of articles and methods used can be seen in the graph shown in Fig. 3, where we present a Venn diagram of methods, their intersection and each paper in a corresponding circle.

5.1.1. Hash-based methods

Hash-based methods employ techniques for hashing claims about the subject in the credential. When sending the credential, the user sends hashed credential claims alongside the values of those they want to disclose. The verifier hashes the values to check if the hashes match. Methods based on hash values include hidden commitment schemes, Merkle hash trees and hashed message authentication codes (HMAC).

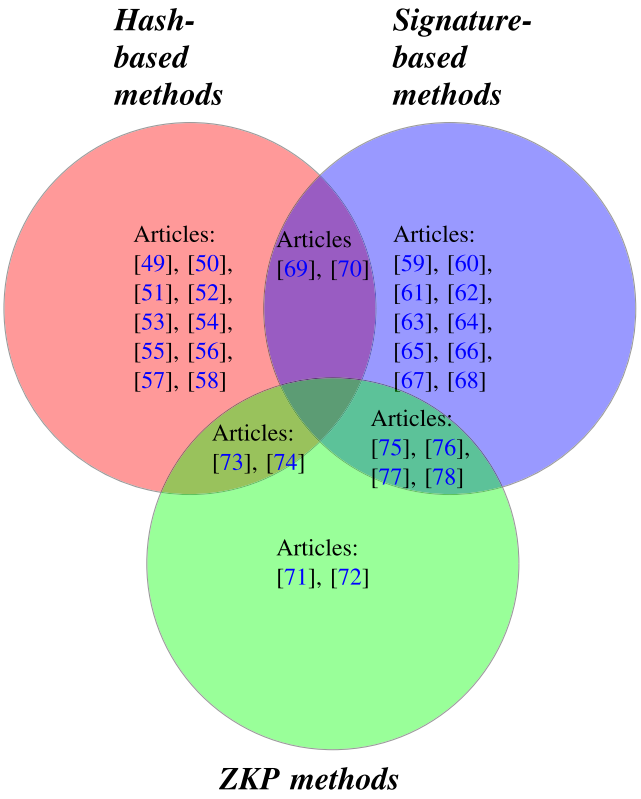

Fig. 3. Venn diagram of methods for selective disclosure.

Message authentication codes. Message authentication code (MAC), also known as an authentication tag, is defined in [49] as "a short piece of information used for authenticating a message. In other words, it confirms that the message comes from the stated sender and has not been changed". Authors of [50] use HMAC (or keyed-hash message authentication code or hash-based message authentication code), a specific type of message authentication code that uses a hash function and a secret key [51]. Using such key derivation, they achieve computational complexity for selective disclosure proportional to the number of disclosed attributes rather than all attributes. Authors of [52] also use keying hash functions for generating hashed VCs and for selective disclosure in the following manner:

1. Issuer prepares credentials $C = claim_1 : hash_1, \ldots, claim_n : hash_n$ using hashed values $hash_i = HMAC(H, key_i, val_i)$, where H is the hash function, key_i a random key, and val_i is the value of the claim. The issuer sends hashed VCs $VC_h = (C, M_{VC}, P_{VC})$, which consists of hashed claims, generated proof through signing P_{VC}, and metadata M_{VC}, which contains digital identifier (DID) issuance and expiration dates, and related attribute data.
2. The subject discloses the desired attributes to the verifier in a hashed VP: $VP_h = (VC_h, M_{VP}, P_{VP})$, which contains hashed VC, presentation metadata M_{VP} with triples of disclosed attributes $(path(claim_i, key_i, value_i)$ and proof P_{VP} signed with the private key.

3. The verifier checks the proofs for VC and VP. For each revealed attribute, the verifier checks the hashed HMAC values.

Hidden commitment schemes. A commitment scheme represents a cryptographic primitive that allows one to commit a chosen value or statement while keeping it hidden from others, with the ability to reveal the committed value later [53]. In the schemes, the user cannot change the committed value, which is binding. They are applicable in several cryptographic protocols and, as such, are underlying in multiple solutions mentioned in this survey. In hiding commitment schemes, the commitments on values are created using hash functions H. The input of these functions is the concatenated values v and "salt" s, i.e. $c = H(v \parallel s)$. Only someone who knows both can open the commitment and prove the value is committed.

In papers [54,55], the authors present an approach to conducting identity-based negotiations. They developed a Trust-χ framework for trust negotiation, providing an XML-based language and a system architecture using digital credentials. Their idea for selective disclosure is defined and implemented as follows:

1. The requester creates its credential C with a list of attribute-value pairs: $C\{(A_1, v_1), \ldots (A_n, v_n)\}$. They send their credential to a trusted third party — the authorizer.
2. The authorizer generates random values and computes commitments using the publicly known hash function and random values r_i: $c_i = H(A_i \| v_i \| r_i)$. They sign the committed credential $C = (c_1, c_2, \ldots, c_n)$ using standard public key infrastructure (PKI) scheme and they get a signature σ. The pair $CCert(C, \sigma)$ is a committed certificate.
3. Authorizer sends the committed certificate $CCert$, with random values r_i, to the requester.
4. The requester and service provider agree on the required attributes. The requester sends the $CCert$, values of attributes they want to reveal $\{(A_i, v_i), \ldots, (A_j, v_j)\}$ and random values $r_i, \ldots . r_j$ to the service provider.
5. The service provider can verify the signature of the authorizer due to the PKI and the values committed using revealed values and random values.

Authors of paper [56] introduce and formally define polynomial commitment schemes $\phi(x)$ over a bilinear pairing group. They provide the way of their construction of $PolyCommit_{DL}$ and show how it can be used for pseudonymous ACs. In their scheme, the committer can efficiently open the commitment to any correct evaluation $\phi(i)$ along with an element called witness w_i. This allows a verifier to confirm that $\phi(i)$ is indeed the evaluation at i of the polynomial $\phi(x)$. Their construction is based on an algebraic property of polynomials $\phi(x) \in \mathbb{Z}_p[x]$ that $(x - i)$ perfectly divides the polynomial $\phi(x) - \phi(i)$. For example, for selective disclosure in credentials, the issuer issues the user with attributes $(m_1, m_2, \ldots, m_n)$, a signed credential (C, σ) that was created using $PolyCommit_{DL}$ where for polynomial ϕ we have the commitment $\phi(i) = m_i$. When the user wants to disclose, for example, a single attribute m_j, they send $(C, \sigma, (j, m_j, w_j))$, which enables verifying a credential and a singular attribute.

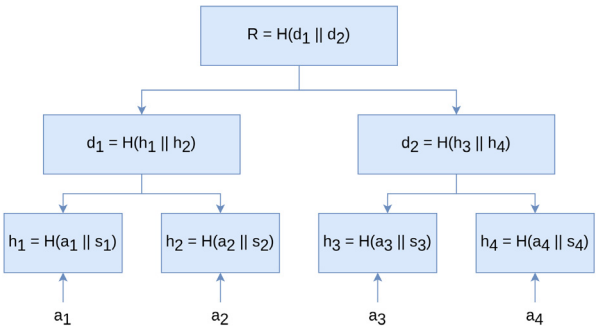

Fig. 4. Merkle tree containing cryptographic hashes of credentials.

In paper [57], authors use soul-bound tokens (SBT) [58], a non-transferable non-fungible token (NFT) [59] representing commitment, credential and affiliation, as a VC. The idea lies in hashing the claims in credentials. Holders have complete control over the information they want to share. The credential metadata is stored on IPFS (Interplanetary File System) [60] encrypted by the holder's public key and can only be seen by the holder. The holder first decrypts the attributes of the credential and then encrypts those they want to reveal with the verifier's public key and sends them. The verifier checks the authenticity by checking the hash with the holder's public key of the information they decrypt with the one stored on the IPFS.

Authors of [61] develop an anonymous certification system where credentials are used to get a service. In this scheme, values of attributes are hashed in the credential. Users show the selectively disclosed values, and for the hidden ones, they calculate an accumulator that they send to the verifiers.

Merkle (hash) tree. A hash tree or Merkle tree is a tree in which every leaf or node is labeled with the cryptographic hash of a data block, and every node that is not a leaf (called a branch, inner node, or inode) is labeled with the cryptographic hash of the labels of its child nodes. A hash tree allows efficient and secure verification of the contents of a large data structure [62]. The idea behind selective disclosure using Merkle trees is similar to the previously mentioned usage of hashing techniques. In Fig. 4, the representation of credentials using the Merkle tree is shown. Hashes for each attribute are calculated $h_i = H(a_i \parallel s_i)$ and stored in leaf nodes. The root of leaves is calculated by creating a combined hash of two $d_i = H(h_i \parallel h_j)$. The process is continued until the Root hash of the Merkle tree is reached. The issuer generates proof and signs the Root. The holder selectively discloses information by sending an inclusion path; for example, for element a_2 in the figure, the holder sends $[a_2, h_1, d_2]$. The verifier can verify the signature and reconstruct the signed Root without knowing all the attributes/leaves, using the inclusion path.

Authors of [62] propose Cerberus, a blockchain-based accreditation and degree verification system, which uses the

Table 4
Overview of hash-based methods.

Article	Algorithm	Complexity	Performance	Suitability	Static/Dynamic	Size/Overhead
[54] [55] [61] [57]	Hash commitments	Generally low because it involves one hashing operation per attribute Depends on size of credential and on hashing function used	Fast processing and verification	Static datasets where integrity is more important than confidentiality or structured proofs.	Static data	Simple proofs Large in size All hashes or disclosed messages are sent
[56]	Polynomial commitment	Higher than regular commitments Depends on selected polynomials	Slower due to the mathematical operations required for committing and verifying attributes	Ideal for applications that require structured proof (ZKP systems)	Static data	Complex proofs with higher computation costs Disclosed data + calculated commitment are shared
[50] [52]	HMAC (keyed-hash message authentication code)	Low because it is similar to hash commitments Requires key management	Efficient but slower than regular hash due to the key-based operations	Useful for authentication in insecure environments Ensures data integrity and authenticity	Static data	Simple proofs Large in size Added overhead due to key management
[62]	Merkle tree	Building O(n) Updates or proofs O(log n)	Efficient for large datasets Allows partial verification	Useful for application where efficient, incremental updates and verifications are needed	Dynamic data	Proof size grows slower than the dataset $\mathcal{O}(\log n)$
[64]	Merkle B-tree with EC	Higher than standard Merkle tree due to multiple child nodes and added overhead of EC	EC can increase tree construction and update time Faster access for non-sequential data operations	Useful for systems where updates are frequent and there is a requirement for security	Dynamic data	Proof size grows slower than the dataset $\mathcal{O}(\log n)$
[63]	Merkle B-tree with encryption	Similar to standard Merkle Tree with added overhead of encryption (complexity depends on algorithm)	Encrypting can increase time for tree construction, update and verification	Useful for systems where enhanced privacy is needed	Dynamic data	Proof size grows slower than the dataset $\mathcal{O}(\log n)$

Merkle tree as explained above, where the Root is recorded on the blockchain.

Authors of [63] implement their scheme minimal disclosure signature authentication (MDSA), where they use the structure of Merkle tree to generate credentials. Prior to hashing they use encryption on every attribute of the user to prevent brute-forcing the hash values.

In paper [64], authors propose an authenticated data structure (ADS) which integrates erasure coding (EC) and Merkle B-tree (MB-tree) for data minimization. MB-tree works like a B+ tree, that consists of B+ tree nodes which are extended with one hash value associated with every pointer entry. This means that for each leaf of a hashed data chunk (left node), there exists a hashed check chunk (right node) calculated to avoid data leakage and security problems. Check chunk is calculated using EC, the data protection method for solving packet loss problems in network transmissions.

Table 4 gives a detailed overview of previously explained hash-based methods. This overview includes the algorithm used, complexity, performance, when the algorithm is suitable, what kind of data it is for, and the size/overhead. It should be noted that exact parameters depend on the hashing algorithms used and that methods should be chosen depending on the need.

5.1.2. Selective disclosure signatures

To achieve selective disclosure in credentials, some authors use selective disclosure signatures. It is possible to selectively disclose claims using these signatures while preserving the ability to verify them.

Blind signatures. Chaum introduced blind signature schemes in [10], the most common cryptographic primitive used alongside commitment schemes. Blind signatures are digital signature schemes where the content is blinded before it is signed. Built upon the introduced blind signatures, Brands defined a Brands credential scheme in [3], where the same credential, signatures and parameters are used in each instance of the showing protocol, which results in a single-show credential system. These signatures are used in practice in U-Prove.

Elliptic curve cryptography. ECC is based on the works of Koblitz [65] and Miller [66] in the 1980s. ECC is an alternative technique to RSA (Rivest–Shamir–Adleman) for public key encryption, which relies on the mathematics of elliptic curves. The mathematical operations are defined over the elliptic curve $y^2 = x^3 + ax + b$ where $4a^3 + 27b^2 \neq 0$. Each value for a and b gives a different curve. The public key is a point on the curve, while the private key is a random number. The public key is obtained by multiplying the private key with a generator point G on the curve. The security depends on the difficulty of the Elliptic Curve Discrete Logarithm Problem (ECDLP).

Let P and Q be two points on the curve such that $kP = Q$, where k is the scalar and elliptic curve discrete logarithm of Q to the base P. Authors of [67] use a generalization of the ECDL function, called ECDLREP function, which is defined as $H = x_1P_1 + x_2P_2 + \cdots + x_lP_l$ for signing a credential. The idea lies in revealing specific attributes and for hidden ones to compute the value that can be used to verify a signature using the specified formula. In paper [68], authors use the Edwards curve, which is known to be among the fastest elliptic curves used to implement cryptographic protocols. The curve can be mathematically expressed as follows: $E_d : x^2+y^2 = 1+dx^2y^2$ where $d \notin \{0, 1\}$. The basis for selective disclosure remains the same.

Group signatures. In the category of group signatures, multiple solutions were proposed, that use CL signatures, BLS signatures and BBS+ signatures.

Jan Camenisch and Anna Lysyanskaya developed CL signatures. Selective disclosure is achieved in the following manner, using CL signatures:

1. Using two random numbers (private keys) p and q, the issuer calculates $n = pq$ and generates $L + 2$ random numbers from a quadratic residue modulo n: $a_1, a_2, \ldots, a_L, b, c \in QR_n$. These random variables with n form the public key.
2. Issuer generates a random prime e and the random number s. Issuer signs the messages $(m_1, \ldots, m_l)$ with signature (e, s, v) where they calculate $v^e = a_1^{m_1} \ldots a_L^{m_L} b^s \; cmodn$.
3. To selectively disclose, the user sends messages they want to reveal, and for those that need to remain hidden, the send a commitment $a_i^{m_i} \; modn$.
4. The verifier verifies the signature by proving that the equation for forming a signature is valid.

CL scheme relies on the strong RSA assumption, requiring long keys and signatures, which results in slow cryptographic operations. The authors of [69] expanded on the CL scheme by defining efficient attributes by encoding discrete binary and finite-set values as prime numbers, which enables the usage of AND, OR and NAND operations.

The BLS signature scheme uses bilinear pairing for verification, and signatures are elements of an elliptic curve. BLS signatures are aggregable, i.e. multiple signatures generated under multiple public keys for multiple messages can be aggregated into a single signature. In [70], authors use the aggregatable property of BLS signatures to form selective disclosure. Each claim of a credential is signed. When selectively disclosing, only the signatures of the disclosed claims are aggregated. The verifier verifies the aggregated signature of the disclosed values and the entire credential.

BBS+ signature relies on the q-Strong Diffie Hellman assumption with pairing-based elliptic-curve cryptography. It requires much shorter keys and signatures. Selective disclosure is achieved through revealing selected attributes and computing signature proof of knowledge (SPK). The signature of the credential is not revealed, and there are two subproofs inside SPK. The first proves the validity of the signature, and the second proves the validity of hidden attributes.

Authors of [71] propose Linked-Data-based VCs that can perform selective disclosure free from the previous scheme's restrictions and prove its property. They propose a method combining multiple credentials issued by different users and how to perform selective disclosure on the set of credentials.

Malleable signatures. Malleable signatures are defined as blank or redactable signatures [72].

Authors of [72] propose an unlinkable redactable signature (URS), in which one redacts message-signature pairs and reveals what they want. They construct an efficient URS scheme using vector commitments and structure-preserving signatures (SPS). SPS are signature schemes where messages, signatures and public keys all consist of elements of a group over which a bilinear map is efficiently computable [73]. To achieve selective disclosure, for a vector of messages $\vec{m} = (1, m_1, \ldots, m_n)$ they create a quote of messages $\vec{m_I} = (2, m'_1, \ldots, m'_n)$, where a quote is a revealed message or $\perp$ if the message is not disclosed. ZKP of signature is used to prove that the messages are from the correct credential.

In paper [74], authors introduce an aggregate signature with randomizable tags for ACs. For each disclosed value, they aggregate signatures and use the ZKP of signature for further verification.

Authors of [75] define a signature scheme for double issuer-hiding attribute-based credentials (DIHAC) built using tag-based aggregatable mercurial signatures (TAM-Sign). These signatures are combined with the structure-preserving signatures on equivalence classes (SPS-EQ) scheme and ZKP of signature. SPS-EQ [76] can randomize both the signed message and the corresponding signature simultaneously. The idea for TAM-Sign lies in aggregating signatures of the same tag for attributes into one compact signature. The scheme also transforms a signature into a new unlinkable signature for the same tag attributes. During selective disclosure, signatures of disclosed attributes are aggregated and converted, and ZKP of signatures is used.

In paper [77], authors build upon the Coconut scheme [78] with threshold issuance (selective disclosure credential scheme), replacing ZKP with polynomial-based unlinkable redactable signature schemes. Using this scheme, they reduce the computation time necessary for selective disclosure. During selective disclosure, a derived signature is computed for disclosed messages.

Authors of [79] introduce permissioned redactable credentials (PERCE), focusing on fine-grained supervision and the membership period. They combine the usage of redactable signatures but in the framework of BLS signature. The user discloses the attributes that they choose and computes redacted signatures for the hidden ones. The verifier can verify the signatures and validity of attributes using these redacted signatures and the original signature for credentials.

Table 5 gives a detailed overview of previously explained signature-based methods. This overview includes the algorithm used, complexity, performance, when the algorithm is suitable, key size and signature size. It should be noted that exact

Table 5
Overview of signature-based methods.

Article	Algorithm	Complexity	Performance	Suitability	Key size[a]	Signature size[a]
[69]	CL signature	High due to the use of interactive ZKP of signatures	Relatively slow due to the complex arithmetic	Suitable for systems that require anonymity features	256 bytes	Can be in kilobytes
[67]	ECDLREP function	Moderate complexity	Efficient due to the properties of elliptic curves	Suitable for systems where performance and compact signatures are required	32 bytes	64 bytes
[72]	URS (SPS signatures)	Moderate to high (depends on specific construction)	Efficient in protocols that need to maintain structure of the message (ZKP)	Used in advanced systems where preserving message is crucial	32 bytes	Can be in kilobytes
[68]	Edwards curve	Low in context of other elliptic curves due to the simpler formulas	Faster calculation and better security	Commonly used in systems like EdDSA	32 bytes	64 bytes
[70]	BLS signature	High due to the use of pairing based cryptography	Signature generation is slower, verification can be fast and aggregation can be done effectively	Particularly useful where aggregation of signatures is needed	48 bytes	96 bytes
[71]	BBS+ signature	High due to the use of pairing based cryptography	Similar to BLS, but with more flexible signatures and message management	Suitable for multi-message systems	96 bytes	112 bytes
[74]	Aggregate signatures with randomizable tags	High due to integration of randomizable tags	Efficient in scenarios where aggregation and randomization are needed simultaneously	Suitable for systems where reusability of signatures without linkability is needed	32 bytes	Can be in kilobytes
[79]	Redactable signatures	High due to the modifying or redacting of signatures	Typically slower due to the additional data management requirements.	Ideal for systems where document integrity is important, especially with authorized edits.	32 bytes	Can be in kilobytes
[77]	Unlinkable redactable signature schemes	Very high due to the combination of unlinkability with redaction	More complex and slower	Ideal for highly sensitive environments redaction	2048 bits	Can be in kilobytes
[75]	Tag-based aggregatable mercurial signatures	Extremely high with the combination of mercurial signatures and tag aggregation	Slower	Suited for systems with complex workflows	2048 bytes	5056 bytes

[a] Depends on the chosen primitive.

parameters, especially key and signature sizes, depend on chosen cryptographic primitives and that sizes mentioned in the table are the more common ones.

5.1.3. Zero-knowledge proof

Constructions that use ZKP use Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge (zk-SNARK). Non-interactive zero-knowledge proofs (NIZK) are cryptographic primitives where the prover can authenticate information between a prover and a verifier without revealing any specific information beyond the statement's validity. This function of encryption makes direct communication between the prover and verifier unnecessary, effectively removing any intermediaries [80]. "Succint" ZKPs can be verified within a few milliseconds, with a proof length of only a few hundred bytes, even for statements about programs that are very large [81].

Authors of [82] design and implement a non-interactive, privacy-preserving credential using ZKPs called ZKlaims. ZKlaims credential $C = (\vec{a}, \vec{y}, S)$, which contains a private bit vector $\vec{a}$, a public bit vector $\vec{y}$ that contains hashes of $\vec{a}$. They first set up a constraint system ϕ, a system of linear constraints that translates into circuits by the zk-SNARK scheme and can support proof of credentials. The constraint system generates proving key pk and verification key vk. To generate proof π, the prover must supply input vectors $\vec{x} = \vec{y}|\vec{p}|\vec{r}$ (where $vecp$ contains predicates, and $\vec{r}$ references) and $\vec{a}$ and proving key pk. A verifier uses a verification key and the public input vector in the procedure, resulting in TRUE or FALSE.

Authors of [83] use pairing-based pre-processing or universal string-based zk-SNARKS as building blocks for VCs. In their scheme, a holder with private user claim data commits the user claim data using commitment schemes. When the issuer receives the values, they authenticate the private values and check the correctness of the commitment. The issuer signs the valid commitments and issues the commitment and its signature as VC. The VC is then recorded in the verifiable data registry. When the holder needs to present their qualification, they generate a minimized public user claim data from their

Table 6
Overview of ZKP.

Article	Type of algorithm	Algorithm used	Characteristic[a]
[82]	zk-SNARK	Groth16	Proof size is among smallest
[83]			Fast to verify
			Requires trusted setup for each proof

[a] In comparison to other zk-SNARK algorithms.

Table 7
Overview of combined approaches.

Article	Category	Algorithms
[85]	ZKP &	Merkle tree + zk-SNARK
[86]	Hash-based	Merkle tree + zk-SNARK + Poseidon hash
[87]		Blind signatures + ZKP
[78]	ZKP &	Threshold signatures + NIZK
[88]	Signature-based	PS-MS + NIZK
[89]		SPS-EQ + NIZK
[90]	Hash-based &	Merkle tree + redactable signatures
[91]	Signature-based	BLS + hash commitments

private claim data directly and guarantee the correctness of it by presenting it with zk-SNARKs proof. The verifier verifies if the public user claim data is derived from the one registered as a commitment by verifying the zk-SNARKs proof. The authors use Commit-and-Prove zk-SNARKS, which allows the verifier to verify the commitment correctness apart from the circuit; the circuit checks only the equality of the commitment in the circuit while verifying it outside the circuit.

Table 6 gives an overview of the algorithm and performance of ZKP-based methods explained above. Both methods use a variant of zk-SNARKs called Groth16, but other variants can be used. Groth16 enables a quadratic arithmetic program to be computed by a prover over elliptic curve points derived in a trusted setup and quickly checked by a verifier [84]. Depending on the type used, different performances are achieved.

5.1.4. Combination of methods

Certain authors combine specific methods to achieve additional privacy-enhancing features. An overview of the mixed methods is shown in Table 7 and explained later in text. Each method inherits its complexity, performance and size, therefore combining methods does not reduce overhead or complexity.

ZKP and signatures. Authors of [78] introduced Coconut, a selective disclosure credential scheme. The idea is based on a threshold issuance signature scheme that allows partial claims to be aggregated into a single credential. Any user may send a request command to a set of signing authorities. This command specifies a set of public or encrypted private attributes that must be certified into the credential. Each authority delivers a partial credential. Users can collect a threshold number of shares, aggregate them to form a single consolidated credential and re-randomize it. The verifier also needs to collect and aggregate authorities' verification keys. The user generates a ZKP for attributes, and the verifier can check the signatures into which the attributes are embedded and if they satisfy the required predicate. This scheme uses threshold issuance, which affects the process of selective disclosure. It uses ZKPs to avoid revealing information, and as such, it falls into the category of combined methods.

In [88], authors combine Pointcheval–Sanders Multi-Signatures (PS-MS) based on a PS pairing-based signature scheme. Their process combines the aggregation of public key shares into a single public key that allows verification of a selectively disclosed credential with ZKP of selectively disclosed value.

Authors of [87] suggest the usage of vector-commitments expanded with the zero-knowledge proof of knowledge (ZKPK) protocols Create, Get, and Set for creating a commitment and retrieving and updating some of the values, with the usage of blind signatures. They call their scheme P-commitments. The authors created two construction forms that rely on the Diffie–Hellman Exponent (DHE) and Computational Diffie–Hellman (ICDH) assumption and compared them. They use P-commitments for privacy-friendly access control, where the service provider only needs information that some user was granted access. A drawback of their solution is that it is a one-time show and requires re-issuing.

In [89], authors combine SPS-EQ and fully adaptive NIZK (AND and NAND proofs) to achieve selective disclosure. They achieve signer-hiding as well, and they define how to obtain mercurial signatures.

ZKP and hash-based approaches. The authors of [85] present a zkCert certification system for digital credentials. It uses smart contracts with Merkle trees to ensure scalability and zk-SNARKs for privacy. The credentials are stored as a Merkle tree, as described above. Hashing is done using Poseidon hash, an elliptic curve hashing algorithm that compresses values into points on the curve, which can be produced and verified efficiently using arithmetic circuits. It is the most efficient hashing function for ZKP applications [92]. Issuance, approval and verification use zk-SNARK circuits as mentioned above.

Authors of [86] use the same combination of Merkle tree, Poseidon hash and zk-SNARKs. However, they add additional leaves to the Merkle tree. The right half of the Merkle tree represents the attributes. The left half represents corresponding metadata (unique identifier, reference to the schema, reference to the revocation registry, public key, expiration date and similar). Their idea is to use zk-SNARKs where applicable and to rely on the Merkle tree for selective disclosure in constrained devices.

Signature and hash-based methods. Authors of [90] introduce the CredChain architecture, which combines a hash-based method and redactable signatures. Credentials are represented through Merkle trees. The redactable signature consists of a signature and auxiliary information about redaction. During the redaction process, hidden attributes are hashed, and auxiliary information is updated in the redactable signature.

In [91], authors propose using the BLS signature due to the aggregation, but they combine it with credentials in which

Table 8
Methods, credentials, ZKP and blockchain in years.

Method	Paper	Year	Credential type	ZKP	Blockchain
Hash-based	[54]	2007	Digital credential		
	[55]	2008	Digital credential		
	[56]	2010	Digital credential	✓	
	[61]	2017	ABC		
	[50]	2019	Digital credential		✓
	[52]	2022	VC		✓
	[63]	2022	Digital credential	✓	✓
	[64]	2023	VC		✓
	[62]	2023	Digital credential		✓
	[57]	2023	SBT		✓
Signature-based	[69]	2008	AC	✓	
	[67]	2009	Digital credential		
	[72]	2015	AC	✓	
	[68]	2019	ABC	✓	
	[70]	2020	AC	✓	
	[71]	2022	VC	✓	✓
	[74]	2023	ABC	✓	
	[79]	2023	AC		✓
	[77]	2023	ABC	✓	✓
	[75]	2023	AC	✓	
ZKP	[82]	2019	ABC	✓	✓
	[83]	2021	VC	✓	✓
ZKP & Signature-based	[87]	2013	AC	✓	
	[78]	2018	ABC	✓	✓
	[88]	2021	PABC	✓	
	[89]	2022	ABC	✓	
ZKP & Hash-based	[85]	2023	VC	✓	✓
	[86]	2023	AC	✓	✓
Signature-based & Hash-based	[90]	2020	VC		✓
	[91]	2022	VC		✓

attributes are hashed values of claims and DID of the user. Attributes are revealed through their values and verified using aggregated signatures and the calculated hash.

5.2. RQ2: Are there differentiations between selective disclosure methods used depending on the digital credential definition or format?

In order to answer the second research question, we present Table 8 of methods and credentials used, together with the year of publishing, and whether the solution uses ZKP and blockchain.

In Section 2, we introduced a historical overview of different types of credentials. Different methods are used for each credential introduced in the section. Even though most of the credentials represent the same thing, specific features are achieved in one or the other. Fig. 5 summarizes the information about the usage of different types of methods on credential types over the surveyed time period. In Fig. 5 and in Table 8, we see that until 2020, the focus was on digital credentials of any kind, especially ACs and ABCs where hash-based and signature-based methods were mostly used. From 2020 and onwards, the focus is on VCs and methods that are used in combination with ZKP.

Table 9 shows an analysis of credential types. For each credential type, the most common algorithm type is defined, whether ZKP and blockchain are commonly used with it, examples in practice, maturity, encoding formats, and characteristics that define the format and explain the advantages and disadvantages of each one.

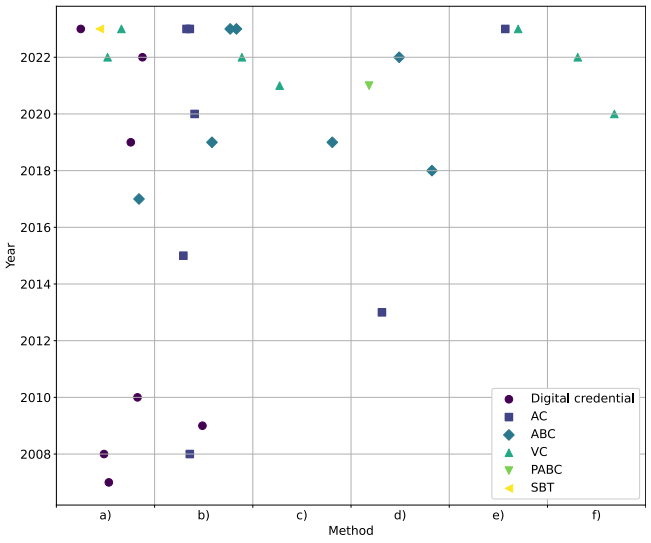

Fig. 5. Different credentials and methods used for selective disclosure throughout the years.
Approaches: (a) Hash-based; (b) Signature-based; (c) ZKP; (d) ZKP & Signature-based; (e) ZKP & Hash-based (f) Signature- & Hash-based;.

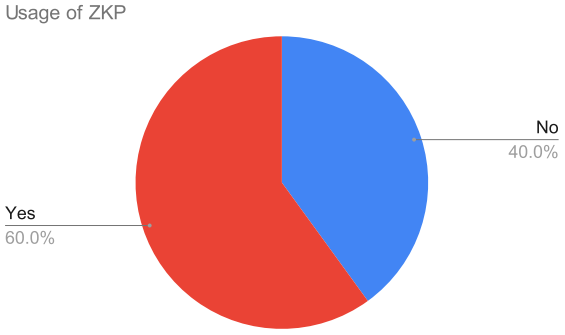

Fig. 6. Number of papers (not)using zero-knowledge proof.

5.3. RQ3: Which methods use zero-knowledge proof?

As described in Section 5, we cannot discuss selective disclosure without mentioning ZKP due to its inherent ability to achieve selective disclosure. As such, there are complete ZKP solutions in some papers, and in others, authors might use ZKP functionalities for certain things, such as verifying signatures without revealing them or showing possible extensions of selective disclosure using bulletproofs [93] or hash-wires [94]. Using bulletproofs, the verifier verifies that a commitment $C(x, r) = xH + rG$ contains a number x. This ZKP method primarily focuses on range proofs, as it was developed for cryptocurrency transactions. Hashwires are used to perform inequality tests and range proofs on committed hashes. As shown in Fig. 6, more than half of the articles include ZKP in their work. Even though more than half of the articles use ZKP, only those solutions that use the ability to hide attribute values are considered ZKP methods for selective disclosure.

ZKP used as a selective disclosure method or in combination with other methods complements selective disclosure in the following manner [95]:

Table 9
Comparison of different credential types.

Type	Algorithm[a]	ZKP[a]	Blockchain[a]	Examples	Maturity	Encoding	Characteristics
Digital credential	Hash			/	/	XML, JSON, PDF, blockchain-based formats, cryptographic tokens, smart contracts	Electronic versions of paper credentials. Any form of digital certification. Easily shareable, verifiable online and can improve administrative efficiency. Focused on transparency and traceability. More general and not inherently designed for privacy enhancement, unless otherwise specified.
AC	Signature	✓		/		JSON, XML, cryptographic tokens	Designed for anonymity of user. Enhances privacy and security by preventing user tracking and profiling. Complex in implementation. Misuse in avoiding accountability possible. ZKP enhancements and signatures can be computationally intensive. Extended versions more commonly used in practice.
ABC	Signature	✓		Idemix, U-prove	IBM, Microsoft, ABC4Trust, PrimeLife	JSON, XML, cryptographic tokens	Extension of ACs focused on attributes. Offers fine granularity over attributes disclosed. Increases user control and enhances privacy. Can be less efficient in terms of computation and storage. Flexibility requires strict policy enforcement mechanisms. Implemented and standardized through extensive work on it.
PABC	ZKP & Signature	✓		/	/	JSON, cryptographic proofs	Privacy enhancement of ABCs through the use of ZKPs. Maximizes privacy by ensuring minimal data exposure. Increases complexity and computational costs are higher. Lack of standardizations and practical usage.
SBT	Hash		✓	/	/	Smart contracts, token metadata	Lack of standardization and practical usage. Reliable and immutable proof of attributes. Depends on blockchain which can cause scalability issues. Non-transferability enhances security but causes lack of flexibility and is restrictive.
VC	All	✓	✓	Hyper-Ledger AnonCreds SD-JWT, Multiple wallets	W3C VC	JSON, JSON-LD, JWT, JWP	Standardized format. Credentials can be independently verified (without direct access to the issuer). Highly interoperable and secure. Enhances trust and reduces fraud. Complex in implementation. Needs widespread adoption of the standard.

[a] Common use.

- Granular control over data sharing — necessary attributes are proven without being revealed;
- Enhanced privacy — sensitive information does not need to be revealed, which reduces the risk of data exposure and lowers the chances of identity theft and fraud since the actual information is not disclosed, just proven;
- Increased trust — allowing the user to control the revealed or proven data, the level of trust increases among users;
- Post-quantum security — certain methods are resistant to quantum attacks;
- Compliance and regulatory adherence — using ZKP compliance and identity can be proved without compromising personal data privacy, as demanded by GDPR or CCPA.

Even though ZKP adds benefits to selective disclosure, it comes with certain disadvantages [95]:

- Complexity — creation and verification with ZKPs can be computationally intensive, which is a problem for systems not optimized for such operations;
- Issues of understanding — designing systems that use ZKPs correctly requires cryptographic expertise, which can be a barrier to widespread adoption;
- Scalability — since ZKPs are computationally intensive, scaling them for large scaling applications remains a challenge;
- Interoperability — integration of ZKP within existing infrastructure is often complex;
- Trusted setup and auditing — some methods require a trusted setup phase;
- Privacy vs. regulation — a balance must be achieved for when ZKP is needed and when it is not.

5.4. RQ4: Which methods are built on blockchain?

With the development of blockchain and the moving of digital identity discussion towards decentralized identity usually built on blockchain, it is interesting to consider whether authors used blockchain in their solutions. Those that came before blockchain did not include it, but starting in 2018, an increasing number of authors have used blockchain as part of their identity solution, which includes selective disclosure. In Fig. 7, we see that the percentile usage of blockchain in the proposed solutions is 50%. Solutions that use blockchain mainly focus on VCs, but there are those that propose the use of blockchain for ACs and ABCs. *Usage of blockchain does not improve on selective disclosure but rather shows us the future trends for developing digital credentials.*

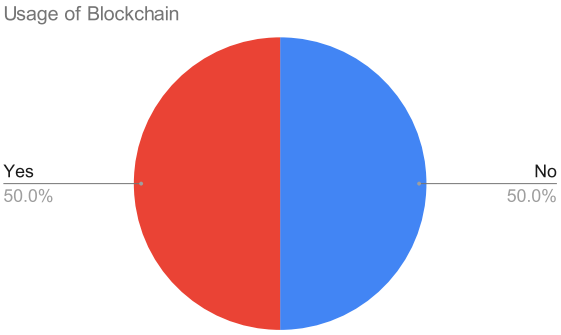

Fig. 7. Usage of blockchain.

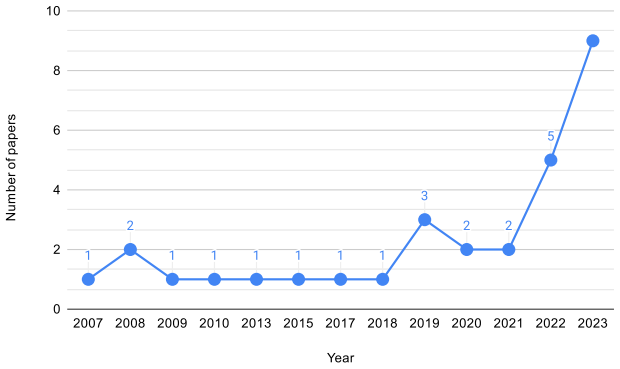

Fig. 8. Distribution of papers by year.

The use of selective disclosure enhances privacy in digital credentials and identities. In digital identity implementations, further enhancements to security and privacy are achieved by using blockchain in the following manner [96]:

- Enhances security and privacy — provides a secure and tamper-proof environment for storing data;
- Full control over identity — blockchain empowers users to have complete control over identity, manage it and share it without intermediaries;
- Inclusion and accessibility — digital identities can provide official identification to individuals who lack traditional identity documents;
- Data integrity and immutability — blockchain is immutable, and it provides a reliable and tamper-resistant source of information;
- Elimination of identity fraud — the risk of identity theft and fraud is reduced;
- Decentralization — removes the risk of a single point of failure;
- Builds trust — approach is user-centric, enhancing user experience and building trust.

Blockchain in digital identity has its drawbacks, especially if we consider integration with selective disclosure [96]:

- Scalability and performance — blockchain is a resource-intensive technology. Combined with computationally intensive methods for selective disclosure, integration can lead to challenges in scalability and performance;
- Complexity in implementation — both blockchain and selective disclosure require significant technical expertise;
- Regulatory and legal challenge — blockchain and selective disclosure enable privacy and, therefore, can face strict regulatory scrutiny;
- Interoperability — lack of standardization between selective disclosure methods and blockchain platforms can cause issues in the interoperability of digital identity systems.

6. Discussion

In our analysis, we identified three main approaches for achieving selective disclosure:

1. hash-based;
2. signature-based;
3. zero-knowledge proof.

Each approach has its specific use cases and its advantages and disadvantages. Hash-based methods are focused on showing the disclosed attribute and are generally considered the least computationally intensive of the three. The main disadvantage lies in the hashing technique, or instead, if it uses salt and if that salt is generated correctly. If the salt is not used, or if it is not randomized for each user, then it is possible to break the hashes for discrete or finite values. Signature-based approaches also rely on showing attributes. They are considered much more computationally intense, especially for devices with limited resources. ZKP methods are considered the most computationally intensive of the three but are the only ones that allow for an attribute value to be proved without being revealed. Analyzing the solutions, we see that one can benefit from combining them as shown in the papers [78,85–91], and it should be noted that no solution of the analyzed combines all three of them.

There is no standardized or perfect solution, and a different solution has been adapted for each use case. The number of papers dealing with selective disclosure has grown in recent years, as shown in Fig. 8, presenting the publication years of analyzed papers. Even though it is a simple graph, it demonstrates the growing academic interest in the problem of selective disclosure.

Regarding publishing venues and publishers, we see a distribution of papers presented in conferences, symposia, workshops and journals, and preprints alongside publishers in Fig. 9. We observe that conference papers are more common, especially those in IEEE colloquia.

7. Research gap

In recent years, more focus has been placed on implementing selective disclosure in digital credentials. Although certain methods have existed for many years, the increasing need for privacy has pushed for further development of different selective disclosure methods.

With privacy becoming an important element of everyday online communication, so do the privacy-enhancing mechanisms. The development of GDPR and CCPA shifted the focus

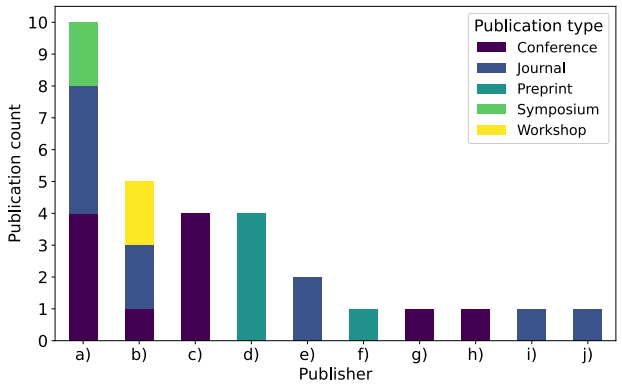

Fig. 9. Publication count by publisher and type.
Publishers: (a) IEEE; (b) ACM; (c) Springer; (d) arXiv preprint; (e) Elsevier Inc; (f) Cryptology ePrint Archive; (g) Dagstuhl Publishing; (h) ITM Web of Conferences; (i) MDPI; (j) Scientific Research.

to privacy-preserving methods such as selective disclosure. Methods for selective disclosure and privacy need to be in concordance with regulations.

In 2023, the Council of the European Union and the European Parliament formalized a provisional agreement on updating and modifying eIDAS (Electronic Identification, Authentication, and Trust Services) Regulation introducing eIDAS2. The agreement focuses on reinforcing security and privacy. It defines regulatory requirements and the need for selective disclosure for convenience and personal data protection, including minimization of processing of personal data [97]. To create a selective disclosure scheme, that will be standardized and adopted in the future, following regulations and recommendations is necessary.

Because of the need for selective disclosure, there are several working drafts for selective disclosure standards defined by European Telecommunications Standards Institute (ETSI) [98] and European Blockchain Services Infrastructure (EBSI) [99]. Methods being developed for selective disclosure should be created in adherence to the regulations and requirements.

As the focus on standardization grows, so do the requirements for selective disclosure schemes. These requirements are [98]:

- Disclosing attributes from at least two separate credentials issued by the same or different issuers;
- Proving disclosed attributes belong to the subject presenting them;
- Ensuring disclosed attributes are unlinkable from multiple presentation sessions;
- Proving that disclosed attributes belong to the appropriate credential.

Some schemes analyzed in Section 5 achieve specific requirements and are focused on them (e.g. unlinkability — ACs and ABCs, combined credentials — BLS signatures), but others do not even consider them. Therefore, there is a research gap in terms of creating a selective disclosure scheme that satisfies all the requirements.

There are several research gaps in terms of implementation:

- Balancing privacy with transparency — one of the biggest challenges is balancing the need for privacy with the requirements for transparency;
- Regulatory compliance — as laws evolve, ensuring that technology complies with international, federal, and national regulations is becoming increasingly complex;
- Security risks — implementing selective disclosure increases the complexity of the encryption system, potentially introducing new vulnerabilities;
- Scalability and efficiency — certain solutions may need to be more scalable and efficient for widespread use. There is a need for a more robust system that can handle large volumes of data.

In recent years, there has been substantial research on quantum computers. If large-scale quantum computers are ever built, they can break many of the public-key cryptosystems currently in use. National Institute of Standards and Technology (NIST) initiated a process to solicit, evaluate, and standardize one or more quantum-resistant public-key cryptographic algorithms, which resulted in CRYSTALS-Dilithium, CRYSTALS-KYBER and SPHINCS+ [100]. As the need for these methods grows, so does the need for their implementation for selective disclosure. Specific methods defined in the previous section are post-quantum secure (specific hash-based methods and ZKP methods), while others are not (specifically signature-based ones); therefore, post-quantum methods should be explored for selective disclosure for future work.

8. Conclusion

This paper presents the most significant articles that address selective disclosure issue. The articles were selected using the Systematic Literature Review [47] methodology. The contributions of this work are as follows:

- Comprehensive literature review: We gave a comprehensive literature review on the broad topic of selective disclosure, identifying seminal works and future trends;
- Differentiation and categorization: We presented a differentiation and categorization of different types and formats for selective disclosure through RQ1. We introduced a new category, ZKP, and showed how combining different methods can improve selective disclosure. We showed the comparative strengths and weaknesses of each selective disclosure method and gave tables to explain the performance of each selective disclosure approach;
- Application across formats: We illustrated through RQ2 how different methods of selective disclosure are applied across various formats. VCs and ACs are currently the most used formats. Hash-based and signature-based methods are the most commonly used approaches for selective disclosure;
- Necessity and benefits of ZKP: We showed how ZKP is necessary for implementing specific signatures, but that is not necessary to achieve selective disclosure through RQ3. Trends suggest that ZKP gives an added benefit to selective disclosure and can be implemented as part of the solution for selective disclosure;

- Future trends in digital identity: We showed that future trends for implementing identity and credentials tend to be focused on using blockchain through RQ4, but that there are benefits and drawbacks in using it;
- Identification of research gaps: We identified critical gaps in current research, from technical to regulative gaps.

Currently, this research area is expanding, and there is still room for improvement for all the defined categories of methods for selective disclosure. There is no clear winner and the "best" universal solution. We encourage researchers to improve on the existing methods, consider new methods or revisit older ones, and even consider methods that are quantum-resistant for the future. The focus should be on finding methods that satisfy all requirements for selective disclosure schemes [98] and specific regulations.

In the future, standardizing credentials will result in interoperable solutions and improve the development of methods for achieving selective disclosure. With this paper, our goal was to create a starting point for researchers interested in achieving selective disclosure in the digital credential world.

CRediT authorship contribution statement

Šeila Bećirović Ramić: Writing – original draft, Investigation, Formal analysis, Conceptualization. **Ehlimana Cogo:** Writing – review & editing, Data curation. **Irfan Prazina:** Writing – review & editing, Methodology. **Emir Cogo:** Writing – review & editing, Visualization. **Muhamed Turkanović:** Supervision, Resources. **Razija Turčinhodžić Mulahasanović:** Supervision. **Saša Mrdović:** Project administration, Funding acquisition, Supervision.

Declaration of competing interest

The authors declare that there is no conflict of interest in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

This work has been supported by Federal Ministry of Education and Sciences of Bosnia and Herzegovina.

References

[1] J. Sedlmeir, R. Smethurst, A. Rieger, G. Fridgen, Digital identities and verifiable credentials, Bus. Inf. Syst. Eng. 63 (5) (2021) 603–613, http://dx.doi.org/10.1007/s12599-021-00722-y.

[2] M. Sporny, M. Jones, T.T. Jr, G. Cohen, I. Herman, Verifiable credentials data model v2.0, 2024, https://www.w3.org/TR/2024/CRD-vc-data-model-2.0-20240513/. (Accessed 19 May 2024).

[3] S. Brands, A technical overview of digital credentials, 2002, https://www.credentica.com/overview.pdf. (Accessed 19 May 2024).

[4] Privacy Patterns, [Support] selective disclosure, 2018, https://www.privacypatterns.org/patterns/Support-Selective-Disclosure. (Accessed: 19 May 2024).

[5] Dock, Selective disclosure guide: Privacy feature of verifiable credentials, 2023, https://www.dock.io/post/selective-disclosure. (Accessed: 19 May 2024).

[6] P. Voigt, A. Von dem Bussche, The eu general data protection regulation (gdpr), in: A Practical Guide, Vol. 10, no. 3152676, 1st ed., Springer International Publishing, Cham, 2017, pp. 10–5555, http://dx.doi.org/10.1007/978-3-319-57959-7.

[7] California Consumer Privacy Act (CCPA), 2018, https://leginfo.legislature.ca.gov/faces/codes_displayText.xhtml?division=3.&part=4.&lawCode=CIV&title=1.81.5. (Accessed: 19 May 2024).

[8] I. Aad, Zero-knowledge proof, in: V. Mulder, A. Mermoud, V. Lenders, B. Tellenbach (Eds.), Trends in Data Protection and Encryption Technologies, Springer Nature Switzerland, Cham, 2023, pp. 25–30, http://dx.doi.org/10.1007/978-3-031-33386-6_6.

[9] S. Nakamoto, Bitcoin: A peer-to-peer electronic cash system, 2008, https://bitcoin.org/bitcoin.pdf. (Accessed: 19 May 2024).

[10] D. Chaum, Blind signatures for untraceable payments, Adv. Cryptol.: Proc. Crypto 82 (1983) 199–203, http://dx.doi.org/10.1007/978-1-4757-0602-4_18.

[11] D. Chaum, Security without identification: Transaction systems to make big brother obsolete, in: Communications of the ACM, vol. 28, (no. 10) ACM New York, NY, USA, 1985, pp. 1030–1044, http://dx.doi.org/10.1145/4372.4373.

[12] F. Baldimtsi, A. Lysyanskaya, Anonymous credentials light, in: Proceedings of the 2013 ACM SIGSAC Conference on Computer and Communications Security, 2013, pp. 1087–1098, http://dx.doi.org/10.1145/2508859.2516687.

[13] I.B. Damgård, Payment systems and credential mechanisms with provable security against abuse by individuals, in: Conference on the Theory and Application of Cryptography, Springer, 1988, pp. 328–335, http://dx.doi.org/10.1007/0-387-34799-2_26.

[14] S.A. Brands, Secret-Key Certificates, CWI (Centre for Mathematics and Computer Science), NLD, 1995, http://dx.doi.org/10.5555/869575.

[15] S. Brands, Restrictive binding of secret-key certificates, in: International Conference on the Theory and Applications of Cryptographic Techniques, Springer, 1995, pp. 231–247, http://dx.doi.org/10.1007/3-540-49264-X_19.

[16] J. Camenisch, A. Lysyanskaya, An efficient system for non-transferable anonymous credentials with optional anonymity revocation, in: Advances in Cryptology—EUROCRYPT 2001: International Conference on the Theory and Application of Cryptographic Techniques Innsbruck, Austria, May 6–10, 2001 Proceedings 20, Springer, 2001, pp. 93–118, http://dx.doi.org/10.1007/3-540-44987-6_7.

[17] J. Camenisch, A. Lysyanskaya, Dynamic accumulators and application to efficient revocation of anonymous credentials, in: Advances in Cryptology—CRYPTO 2002: 22nd Annual International Cryptology Conference Santa Barbara, California, USA, August 18–22, 2002 Proceedings 22, Springer, 2002, pp. 61–76, http://dx.doi.org/10.1145/357353.357357.

[18] J. Camenisch, A. Lysyanskaya, A signature scheme with efficient protocols, in: Security in Communication Networks: Third International Conference, SCN 2002 Amalfi, Italy, September 11–13, 2002 Revised Papers 3, Springer, 2003, pp. 268–289, http://dx.doi.org/10.1007/3-540-36413-7_20.

[19] J. Camenisch, A. Lysyanskaya, Signature schemes and anonymous credentials from bilinear maps, in: Annual International Cryptology Conference, Springer, 2004, pp. 56–72, http://dx.doi.org/10.1007/978-3-540-28628-8_4.

[20] D. Boneh, B. Lynn, H. Shacham, Short signatures from the weil pairing, in: International Conference on the Theory and Application of Cryptology and Information Security, Springer, 2001, pp. 514–532, http://dx.doi.org/10.1007/3-540-45682-1_30.

[21] D. Boneh, C. Gentry, B. Lynn, H. Shacham, Aggregate and verifiably encrypted signatures from bilinear maps, in: Advances in Cryptology—EUROCRYPT 2003: International Conference on the Theory and Applications of Cryptographic Techniques, Warsaw, Poland, May 4–8, 2003 Proceedings 22, Springer, 2003, pp. 416–432, http://dx.doi.org/10.1007/3-540-39200-9_26.

[22] B. Edgington, Upgrading Ethereum: A technical handbook on Ethereum's move to proof of stake and beyond, 2022, https://eth2book.info/capella/. (Accessed: 19 May 2024).

[23] H. Narumanchi, L. Maddali, N. Emmadi, Privacy enabled immunity credential system on blockchain, in: International Conference on Communication Systems and Networks, 2022, http://dx.doi.org/10.1109/COMSNETS53615.2022.9668579.
[24] D. Boneh, X. Boyen, H. Shacham, Short group signatures, in: Annual International Cryptology Conference, Springer, 2004, pp. 41–55, http://dx.doi.org/10.1007/978-3-540-28628-8_3.
[25] M.H. Au, W. Susilo, Y. Mu, Constant-size dynamic k-TAA, in: Security and Cryptography for Networks: 5th International Conference, SCN 2006, Maiori, Italy, September 6-8, 2006. Proceedings 5, Springer, 2006, pp. 111–125, http://dx.doi.org/10.1007/11832072_8.
[26] C. Paquin, U-prove technology overview V1.1 (revision 2), 2013, https://www.microsoft.com/en-us/research/publication/u-prove-technology-overview-v1-1-revision-2/. (Accessed 19 May 2024).
[27] Credentica, Welcome to credentica, 2004-2023, https://www.credentica.com/. (Accessed: 19 May 2024).
[28] J. Camenisch, E. Van Herreweghen, Design and implementation of the idemix anonymous credential system, in: Proceedings of the 9th ACM Conference on Computer and Communications Security, ACM, 2002, pp. 21–30, http://dx.doi.org/10.1145/586110.586114.
[29] U. König, Identity mixer, 2011, http://primelife.ercim.eu/results/opensource/55-identity-mixer. (Accessed: 19 May 2024).
[30] ABC4Trust, ABC4Trust - Attribute-based credentials for trust, 2022, https://abc4trust.eu/. (Accessed: 19 May 2024).
[31] J. Hajny, L. Malina, Unlinkable attribute-based credentials with practical revocation on smart-cards, in: Smart Card Research and Advanced Applications: 11th International Conference, CARDIS 2012, Graz, Austria, November 28-30, 2012, Revised Selected Papers 11, Springer, 2013, pp. 62–76, http://dx.doi.org/10.1007/978-3-642-37288-9_5.
[32] Yivi, Yivi app, 2023, (Accessed: 19 May 2024).
[33] P. Bichsel, J. Camenisch, T. Groß, V. Shoup, Anonymous credentials on a standard java card, in: Proceedings of the 16th ACM Conference on Computer and Communications Security, CCS '09, Association for Computing Machinery, New York, NY, USA, 2009, pp. 600–610, http://dx.doi.org/10.1145/1653662.1653734.
[34] J. Camenisch, A. Lehmann, G. Neven, A. Rial, Privacy-preserving auditing for attribute-based credentials, in: ESORICS 2014: Computer Security - ESORICS 2014, 2014, pp. 109–127, http://dx.doi.org/10.1007/978-3-319-11212-1_7.
[35] M. Yuliana, A. Pratiarso, A. Sudarsono, Proof of attributes based CL signature scheme on e-health applications, in: 2015 International Conference on Science in Information Technology, ICSITech, 2015, http://dx.doi.org/10.1109/ICSITECH.2015.7407813.
[36] P. Conejero-Alberola, M.F. Hinarejos, J.L. Ferrer-Gomila, A selective privacy-preserving identity attributes protocol for electronic coupons, in: Lecture Notes in Computer Science, 2017, http://dx.doi.org/10.1007/978-3-319-93524-9_11.
[37] Hyperledger, Hyperledger, 2019, https://www.hyperledger.org/. (Accessed: 19 May 2024).
[38] K.C. Toth, A. Anderson-Priddy, Self-sovereign digital identity: A paradigm shift for identity, IEEE Secur. Privacy 17 (3) (2019) 17–27, http://dx.doi.org/10.1109/MSEC.2018.2888782.
[39] D. Fett, K. Yasuda, B. Campbell, Selective Disclosure for JWTs (SD-JWT), 2024, https://datatracker.ietf.org/doc/draft-ietf-oauth-selective-disclosure-jwt/08/. (Accessed 19 May 2024).
[40] C. Diaz, H. Halpin, A. Kiayias, The nym network, 2021, https://nymtech.net/nym-whitepaper.pdf. (Accessed 19 May 2024).
[41] R.D. Garcia, G. Ramachandran, R. Jurdak, J. Ueyama, Blockchain-aided and privacy-preserving data governance in multi-stakeholder applications, IEEE Trans. Netw. Serv. Manag. (2022) http://dx.doi.org/10.1109/TNSM.2022.3225254.
[42] I. Sertkaya, P.B. Roenne, Estonian internet voting with anonymous credentials, Turkish J. Electr. Eng. Comput. Sci. (2021) http://dx.doi.org/10.3906/ELK-2105-197.
[43] N. Kaaniche, M. Laurent, S. Belguith, Privacy enhancing technologies for solving the privacy-personalization paradox: Taxonomy and survey, J. Network Comput. Appl. 171 (2020) 102807, http://dx.doi.org/10.1016/j.jnca.2020.102807.
[44] A.A. Corici, T. Hühnlein, D. Hühnlein, O. Rode, Towards interoperable vaccination certificate services, in: Proceedings of the 16th International Conference on Availability, Reliability and Security, 2021, pp. 1–9, http://dx.doi.org/10.1145/3465481.3470035.
[45] R. Mukta, H.-y. Paik, Q. Lu, S.S. Kanhere, A survey of data minimisation techniques in blockchain-based healthcare, Comput. Netw. 205 (2022) 108766, http://dx.doi.org/10.1016/j.comnet.2022.108766.
[46] F. Andrea, S. Ranise, G. Sciarretta, S. Mario, A. Sharif, A. Tomasi, et al., A first appraisal of cryptographic mechanisms for the selective disclosure of verifiable credentials, in: Proceedings of the 20th International Conference on Security and Cryptography SECRYPT-Volume 1, vol. 1, 2023, pp. 123–134, http://dx.doi.org/10.5220/0012084000003555.
[47] B.A. Kitchenham, S. Charters, Guidelines for performing systematic literature reviews in software engineering, 2007, https://www.elsevier.com/__data/promis_misc/525444systematicreviewsguide.pdf. (Accessed 19 May 2024).
[48] A. Sambra, Verifiable credentials implementation guidelines 1.0, 2019, https://www.w3.org/TR/2019/NOTE-vc-imp-guide-20190924/. (Accessed: 19 May 2024).
[49] D. Liu, M. Caceres, T. Robichaux, D.V. Forte, E.S. Seagren, D.L. Ganger, B. Smith, W. Jayawickrama, C. Stokes, J. Kanclirz (Eds.), Chapter 3 - an introduction to cryptography, in: Next Generation SSH2 Implementation, Syngress, Burlington, 2009, pp. 41–64, http://dx.doi.org/10.1016/B978-1-59749-283-6.00003-9.
[50] D.W. Kravitz, Exploration and impact of blockchain-enabled adaptive non-binary trust models, in: ICDCN, 2019, http://dx.doi.org/10.1145/3288599.3288639.
[51] D.H. Krawczyk, M. Bellare, R. Canetti, HMAC: Keyed-Hashing for Message Authentication, 1997, http://dx.doi.org/10.17487/RFC2104, RFC 2104.
[52] A.D. Salve, A. Lisi, P. Mori, L. Ricci, Selective disclosure in self-sovereign identity based on hashed values, in: International Symposium on Computers and Communications, 2022, http://dx.doi.org/10.1109/ISCC55528.2022.9913052.
[53] O. Goldreich, Foundations of Cryptography: Volume 1, Cambridge University Press, USA, 2006, http://dx.doi.org/10.1017/CBO9780511546891.
[54] A. Squicciarini, E. Bertino, E. Ferrari, F. Paci, B. Thuraisingham, PP-trust-x: A system for privacy preserving trust negotiations, in: TSEC, 2007, http://dx.doi.org/10.1145/1266977.1266981.
[55] A. Squicciarini, A. Trombetta, E. Bertino, S. Braghin, Identity-based long running negotiations, in: DIM '08, 2008, http://dx.doi.org/10.1145/1456424.1456440.
[56] A. Kate, G.M. Zaverucha, I. Goldberg, Constant-size commitments to polynomials and their applications, in: International Conference on the Theory and Application of Cryptology and Information Security, 2010, http://dx.doi.org/10.1007/978-3-642-17373-8_11.
[57] S.S. Reddy, D.S. Kushwaha, Framework for privacy preserving credential issuance and verification system using soulbound token, in: ITM Web of Conferences, 2023, http://dx.doi.org/10.1051/ITMCONF/20235606002.
[58] E.G. Weyl, P. Ohlhaver, V. Buterin, Decentralized society: Finding web3's soul, 2022, http://dx.doi.org/10.2139/ssrn.4105763, Available At SSRN 4105763.
[59] Q. Wang, R. Li, Q. Wang, S. Chen, Non-fungible token (NFT): Overview, evaluation, opportunities and challenges, 2021, arXiv:2105.07447.
[60] J. Benet, IPFS - content addressed, versioned, P2P file system, 2014, arXiv:1407.3561.
[61] C. Kiennert, N. Kaaniche, M. Laurent, P.-O. Rocher, J. Garcia-Alfaro, Anonymous certification for an e-assessment framework, in: Lecture Notes in Computer Science, 2017, http://dx.doi.org/10.1007/978-3-319-70290-2_5.
[62] A. Tariq, H. Binte Haq, S.T. Ali, Cerberus: A blockchain-based accreditation and degree verification system, IEEE Trans. Comput. Soc. Syst. 10 (4) (2023) 1503–1514, http://dx.doi.org/10.1109/TCSS.2022.3188453.

[63] Z. Yang, H. Ma, M. Ai, M. Zhan, G. Wu, Y. Zhang, A minimal disclosure signature authentication scheme based on consortium blockchain, in: 2022 IEEE International Conference on Blockchain (Blockchain), 2022, pp. 516–521, http://dx.doi.org/10.1109/BLOCKCHAIN55522.2022.00079.

[64] R. Tian, L. Kong, B. Zhang, X. Li, Q. Li, Authenticated selective disclosure of credentials in hybrid-storage blockchain, in: International Conference on Parallel and Distributed Systems, 2023, http://dx.doi.org/10.1109/ICPADS56603.2022.00050.

[65] N. Koblitz, Elliptic curve cryptosystems, Math. Comput. 48 (177) (1987) 203–209, http://dx.doi.org/10.1090/S0025-5718-1987-0866109-5.

[66] V.S. Miller, Use of elliptic curves in cryptography, in: Conference on the Theory and Application of Cryptographic Techniques, Springer, 1985, pp. 417–426, http://dx.doi.org/10.1007/3-540-39799-X_31.

[67] A. Athavale, K. Singh, S. Sood, Design of a private credentials scheme based on elliptic curve cryptography, in: 2009 First International Conference on Computational Intelligence, Communication Systems and Networks, 2009, http://dx.doi.org/10.1109/CICSYN.2009.80.

[68] I. Sene, A.A. Ciss, O. Niang, I2PA: An efficient ABC for IoT, in: Cryptography, 2019, http://dx.doi.org/10.3390/CRYPTOGRAPHY3020016.

[69] J. Camenisch, T. Groß, Efficient attributes for anonymous credentials, ACM Trans. Inf. Syst. Secur. (2012) http://dx.doi.org/10.1145/2133375.2133379.

[70] A. Rondelet, A note on anonymous credentials using BLS signatures, 2020, arXiv:2006.05201.

[71] D. Yamamoto, Y. Suga, K. Sako, D. Yamamoto, Y. Suga, K. Sako, Formalising linked-data based verifiable credentials for selective disclosure, in: 2022 IEEE European Symposium on Security and Privacy Workshops, EuroSPW, 2022, pp. 52–65, http://dx.doi.org/10.1109/EUROSPW55150.2022.00013.

[72] J. Camenisch, M. Dubovitskaya, K. Haralambiev, M. Kohlweiss, Composable and modular anonymous credentials: Definitions and practical constructions, in: ASIACRYPT 2015: Advances in Cryptology – ASIACRYPT 2015, 2015, pp. 262–288, http://dx.doi.org/10.1007/978-3-662-48800-3_11.

[73] B. Libert, T. Peters, M. Joye, M. Yung, Linearly homomorphic structure-preserving signatures and their applications, Des., Codes Cryptogr. 77 (2015) 441–477, http://dx.doi.org/10.1007/s10623-015-0079-1.

[74] C. Hébant, D. Pointcheval, Traceable constant-size multi-authority credentials, in: International Conference on Security and Cryptography for Networks, 2023, http://dx.doi.org/10.1007/978-3-031-14791-3_18.

[75] R. Shi, Y. Yang, Y. Li, H. Feng, G. Shi, H.H. Pang, R.H. Deng, Double issuer-hiding attribute-based credentials from tag-based aggregatable mercurial signatures, IEEE Trans. Dependable Secure Comput. (2023) http://dx.doi.org/10.1109/TDSC.2023.3314019/mm1.

[76] G. Fuchsbauer, C. Hanser, D. Slamanig, Structure-preserving signatures on equivalence classes and constant-size anonymous credentials, J. Cryptol. 32 (2019) 498–546, http://dx.doi.org/10.1007/s00145-018-9281-4.

[77] R. Shi, H. Feng, Y. Yang, F. Yuan, Y. Li, H. Pang, R.H. Deng, Threshold attribute-based credentials with redactable signature, IEEE Trans. Serv. Comput. 16 (5) (2023) 3751–3765, http://dx.doi.org/10.1109/TSC.2023.3280914.

[78] A. Sonnino, M. Al-Bassam, S. Bano, S. Meiklejohn, G. Danezis, Coconut: Threshold issuance selective disclosure credentials with applications to distributed ledgers, 2020, arXiv:1802.07344.

[79] Y. Liu, D. He, Q. Feng, M. Luo, K. Choo, PERCE: A permissioned redactable credentials scheme for a period of membership, IEEE Trans. Inf. Forensics Secur. (2023) http://dx.doi.org/10.1109/TIFS.2023.3274435.

[80] O. Goldreich, Foundations of Cryptography, Volume 2, Cambridge university press Cambridge, USA, 2004, http://dx.doi.org/10.1017/CBO9780511721656.

[81] D. Hopwood, S. Bowe, T. Hornby, N. Wilcox, et al., Zcash protocol specification, 2024, https://raw.githubusercontent.com/zcash/zips/master/protocol/protocol.pdf. (Accessed: 19 May 2024).

[82] M. Schanzenbach, T. Kilian, J. Schütte, C. Banse, ZKlaims: Privacy-preserving attribute-based credentials using non-interactive zero-knowledge techniques, in: Proceedings of the 16th International Joint Conference on E-Business and Telecommunications SECRYPT - Volume 1, 2019, pp. 325–332, http://dx.doi.org/10.5220/0007772903250332.

[83] J. Lee, J. Choi, H. Oh, J. Kim, Privacy-preserving identity management system, 2021, Cryptology ePrint Archive, Paper 2021/1459. URL https://eprint.iacr.org/2021/1459. (Accessed 19 May 2024).

[84] J. Groth, On the size of pairing-based non-interactive arguments, in: Advances in Cryptology–EUROCRYPT 2016: 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part II 35, Springer, 2016, pp. 305–326, http://dx.doi.org/10.1007/978-3-662-49896-5_11.

[85] R.Q. Saramago, H. Meling, L. Jehl, A privacy-preserving and transparent certification system for digital credentials, in: International Conference on Principles of Distributed Systems, 2022, http://dx.doi.org/10.4230/LIPICS.OPODIS.2022.9.

[86] M. Babel, J. Sedlmeir, Bringing data minimization to digital wallets at scale with general-purpose zero-knowledge proofs, 2023, arXiv:2301.00823.

[87] M. Kohlweiss, A. Rial, Optimally private access control, in: WPES '13: Proceedings of the 12th ACM Workshop on Workshop on Privacy in the Electronic Society, 2013, pp. 37–48, http://dx.doi.org/10.1145/2517840.2517857.

[88] J. García-Rodríguez, R.T. Moreno, J.B. Bernabe, A.F. Skarmeta, Implementation and evaluation of a privacy-preserving distributed ABC scheme based on multi-signatures, J. Inf. Secur. Appl. (2021) http://dx.doi.org/10.1016/J.JISA.2021.102971.

[89] A. Connolly, P. Lafourcade, O.P. Kempner, Improved constructions of anonymous credentials from structure-preserving signatures on equivalence classes, in: Public-Key Cryptography, PKC 2022, 2022, pp. 409–438, http://dx.doi.org/10.1007/978-3-030-97121-2_15.

[90] R. Mukta, J. Martens, H.-y. Paik, Q. Lu, S.S. Kanhere, Blockchain-based verifiable credential sharing with selective disclosure, in: 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom, IEEE, 2020, pp. 959–966, http://dx.doi.org/10.1109/TrustCom50675.2020.00128.

[91] Z. Li, A verifiable credentials system with privacy-preserving based on blockchain, J. Inf. Secur. 13 (2) (2022) http://dx.doi.org/10.4236/JIS.2022.132003.

[92] L. Grassi, D. Khovratovich, C. Rechberger, A. Roy, M. Schofnegger, Poseidon: A new hash function for Zero-Knowledge proof systems, in: 30th USENIX Security Symposium (USENIX Security 21), USENIX Association, 2021, pp. 519–535.

[93] B. Bünz, J. Bootle, D. Boneh, A. Poelstra, P. Wuille, G. Maxwell, Bulletproofs: Short proofs for confidential transactions and more, in: 2018 IEEE Symposium on Security and Privacy, SP, IEEE, 2018, pp. 315–334, http://dx.doi.org/10.1109/SP.2018.00020.

[94] K. Chalkias, S. Cohen, K. Lewi, F. Moezinia, Y. Romailler, HashWires: Hyperefficient credential-based range proofs, 2021, Cryptology ePrint Archive, Paper 2021/297. URL https://eprint.iacr.org/2021/297. (Accessed: 19 May 2024).

[95] L. Zhou, A. Diro, A. Saini, S. Kaisar, P.C. Hiep, Leveraging zero knowledge proofs for blockchain-based identity sharing: A survey of advancements, challenges and opportunities, J. Inf. Secur. Appl. 80 (2024) 103678, http://dx.doi.org/10.1016/j.jisa.2023.103678.

[96] Finance Magnates, Blockchain-based digital identity: Benefits, risks, and implementation challenges, 2023, https://www.financemagnates.com/cryptocurrency/education-centre/blockchain-based-digital-identity-benefits-risks-and-implementation-challenges/. (Accessed: 19 May 2024).

[97] 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, 2021, https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A52021PC0281. (Accessed: 19 May 2024).

[98] S. Elfors, A. Burckard, ETSITR 119 476 v1.1.1:Electronic signatures and infrastructures (ESI)-analysis of selective disclosure and zero-knowledge proofs applied to electronic attestation of attributes, 2023, https://www.etsi.org/deliver/etsi_tr/119400_119499/119476/01.01.01_60/tr_119476v010101p.pdf. (Accessed 19 May 2024).

[99] EBSI, Selective disclosure: An EBSI improvement proposal, 2023, https://ec.europa.eu/digital-building-blocks/sites/display/EBSI/Selective+Disclosure%3A+An+EBSI+Improvement+Proposal. (Accessed: 19 May 2024).

[100] G. Alagic, D. Cooper, Q. Dang, T. Dang, J.M. Kelsey, J. Lichtinger, Y.-K. Liu, C.A. Miller, D. Moody, R. Peralta, R. Perlner, A. Robinson, D. Smith-Tone, D. Apon, Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process, NIST Interagency/Internal Report (NISTIR), National Institute of Standards and Technology, Gaithersburg, MD, 2022, http://dx.doi.org/10.6028/NIST.IR.8413.