Soft Comput (2016) 20:1781–1791
DOI 10.1007/s00500-015-1704-7

FOCUS

Random oracle-based anonymous credential system for efficient attributes proof on smart devices

Nan Guo[1] · **Tianhan Gao**[2] · **Hwagyoo Park**[3]

Published online: 22 May 2015

Abstract Attributes proof in anonymous credential systems is an effective way to balance security and privacy in user authentication; however, the linear complexity of attributes proof causes the existing anonymous credential systems far away from being practical, especially on resource-limited smart devices. For efficiency considerations, we present a novel pairing-based anonymous credential system which solves the linear complexity of attributes proof based on aggregate signature scheme. We propose two extended signature schemes, BLS+ and BGLS+, to be cryptographical building blocks for constructing anonymous credentials in the random oracle model. Identity-like information of message holder is encoded in a signature in order that the message holder can prove the possession of the input message along with the validity of a signature. We present issuance protocol for anonymous credentials embedding weak attributes which are referred to what cannot identify a user in a population. Users can prove any combination of attributes all at once by aggregating the corresponding individual credentials into one. The attributes proof protocols on AND and OR relation over multiple attributes are also given. The performance analysis shows that the aggregation-based anonymous credential system outperforms both the conventional Camenisch–Lysyanskaya pairing-based system and the accumulator-based system when prove AND and OR relation over multiple attributes, and the size of credential and public parameters are shorter as well.

Keywords Privacy · Anonymous credential · Attributes proof · Aggregate signature

Communicated by A. Jara, M.R. Ogiela, I. You, and F.-Y. Leu.

✉ Nan Guo
guonan@mail.neu.edu.cn

Tianhan Gao
gaoth@mail.neu.edu.cn

Hwagyoo Park
hkpark1@sch.ac.kr

[1] Information Science and Engineering College, Northeastern University, Shenyang, People's Republic of China

[2] Software College, Northeastern University, Shenyang, People's Republic of China

[3] Department of Healthcare Administration and Management, Soonchunhyang University, Seoul, Republic of Korea

1 Introduction

In privacy-sensitive applications involving individuals' date of birth, minority and social benefit status, personal healthcare or financial data, the attribute-based authentication and access control is more desirable because attributes are less likely than identifiers to privacy leak. Generally, they are encoded in binary or finite set (Amang et al. 2011; Jan and Thomas 2008) and represented to a group instead of a person. In such group authentication cases, it is important that service providers be able to convince the user has the required permissions for accessing the services, while at the same time the user's attributes be proven in such a fuzzy way that only the minimum amount of necessary attributes to accomplish a certain goal should be collected and the identity of user is kept uncertain as well.

Anonymous credential system has proved to meet such security and privacy requirements (Jan and Anna 2004; Camenisch et al. 2012; Jan and Thomas 2008). It allows users to obtain a credential from an Issuer on a number of attributes and prove the possession of credential to a verifier without revealing any other information about themselves. As an attribute credential, it also enable users to selectively

release and prove a subset of the certified attributes while others are hidden completely. The reason that they have become so popular is that they strictly adhere to data minimization principles: no electronic transaction should require its participants to needlessly reveal private information (Baldimtsi and Lysyanskaya 2012).

Industry aims at employing anonymous credential systems on smart devices with limited computational power. Examples include smart phones and corporate- or government-issued electronic identity cards. For efficiency considerations, the existing anonymous credential systems use either the RSA group or bilinear group, where the security parameters in these groups make the systems expensive for source-limited smart devices. Such efficiency considerations are particularly important when using anonymous credentials to prove attributes which are specified in relying parties' policy. Users can either prove the possession of all of the multiple attributes, i.e., AND relation over attributes, or prove the possession of one of the multiple attributes, i.e., OR relation over attributes (Amang et al. 2011). For example, when submitting a resume, a person's gender has to be a $female$, the nationality is $French$, and the degree is $Ph.D$; while in the other scenario, one person can enjoy the free tickets with his ID-card only if his minority is $blind$ or social_benefit is $unemployed$ or the type is $kids\_card$. Unfortunately, attributes proof in the existing anonymous credentials suffers from linear complexity in the total number of the user's attributes.

The state-of-the-art solution to linear complexity of attributes proof mainly focuses on reducing the number of exponentiations by employing cryptographic accumulators (Amang et al. 2011; Jan et al. 2009). They allow one to hash a large set of inputs in a single short value, the *accumulator*, and then provide evidence by an *accumulator witness* that a given value is indeed contained in the accumulator. A large set of input values can be assigned to finite-set attributes, and an accumulator will output a constant-size value. Thus, multiple attributes can be proved with constant complexity in the number of finite-set attributes. However, the complexity still depends on the number of string attributes, and the size of public keys is depending on the number of attribute values. Even more importantly, the number of extra pairings largely increases for verifying the validity of accumulator, so they are still far away from being practical on smart devices.

To overcome the efficiency constriction of attributes proof, we consider privacy requirements for anonymous credential systems regarding *strong* and *weak* attributes. In Abhilasha et al. (2010), a strong attribute uniquely identifies an individual in a population, whereas a weak attribute can be applied to many individuals in a population. Whether an attribute is strong or weak depends upon the size of the population and the uniqueness of the identity attribute. Examples of strong attributes are a user's passport number or social security number. Examples of weak attributes are age, profession and gender. Generally, the privacy requirements for anonymous credential systems consist of anonymity, unlinkability and selective disclosure of attributes; for not loss of generality, they imply privacy protection on strong attributes. If the privacy requirements could be relaxed in the case of weak attributes, the efficiency of attributes proof will be more increased as a result. Concisely, attributes are generally encoded as discrete logarithm representation in most anonymous credentials (Jan and Anna 2004; Amang et al. 2011; Jan et al. 2009) in order that what is being signed is an information-theoretically secure commitment of attributes instead of the actual value of it. Such encoding method brings much computation cost to users. It is crucial to obtain anonymity when showing a credential embedding strong attributes, but not economical to weak attributes because they are inherently identification-resistant even if the values are revealed in the clear. Therefore, we aim to construct an anonymous credential system particularly for weak attributes. Such idea is novel but reasonable. On one hand, according to the surveyed different data sets for electronic identity cards and driver's license cards in Jan and Thomas (2008), we observe that only a minority of attributes are generic string or integer which generally represent strong attributes, whereas most attributes are either binary or taken from a finite set of discrete values which generally represent weak attributes. On the other hand, in attributes-based authenticate and access control, weak attributes can fully meet relying parties' security policies. Therefore, if we distinguish weak attributes from strong ones when constructing anonymous credentials, the privacy requirements can be relaxed and the efficiency of attributes proof will be improved as a result.

Our contributions are twofold. (1) Present two extended signature schemes, BLS+ and BGLS+, based on Boneh–Lynn–Shacham (short for BLS) short signature scheme (Boneh et al. 2001) and Boneh–Gentry–Lynn–Shacham (short for BGLS) aggregate signature scheme (Boneh et al. 2003). The possession of a input message is considered in such schemes. The identity-like information of message holder is encoded in a signature. Later on, the possession of the input message can be proved along with the validity of a signature. The security of BLS+ and BGLS+ signature schemes is proved whenever input messages to the sign oracle are distinct or not. (2) Construct a pairing-based anonymous credential system particularly for *weak* attributes, and construct attributes proof protocols based on BLS+ and BGLS+ signature schemes, in the random oracle model. Users can prove any combination of attributes all at once by aggregating the corresponding individual credentials into a single one. The advantages of aggregate signature in public key size, signature size and verification efficiency are used to construct

attributes proof over multiple attributes with constant complexity.

This article is a revised and expanded version of Nan et al. (2014). The main difference between this paper and Nan et al. (2014) is that we propose more efficient cryptographical building blocks to construct anonymous credentials, i.e., BLS+ short signature scheme and BGLS+ aggregate signature scheme. Moreover, Nan et al. (2014) is presented in business processes environment, while this paper focuses on solving linear complexity problem of attributes proof on resource-limited smart devices. The raw idea of BLS+ and BGLS+ signature schemes is introduced in Nan et al. (2013), however the security proof is limited; and it just presents the construction of AND relation proof and never mention OR relation proof ever. This paper gives more solid security proof and presents the security provable issuance protocol, AND and OR relation proof protocols as well.

We organize the remainder of the paper as follows. In Sect. 2, we propose BLS+ and BGLS+ signature schemes and prove them secure. In Sect. 3, we introduce weak attributes encoding method and issuance protocol. In Sect. 4, attributes proof protocols on AND and OR relation are given, respectively. In Sect. 5 we analyze the performance of our system and comparison with the conventional Camenisch–Lysyanskaya pairing-based system and the accumulator-based system. Section 6 is the related work on anonymous credential systems and their building blocks. The final section is the conclusion.

2 Preliminary

In this section, we review a few concepts related to GDH groups and bilinear maps (Boneh et al. 2001, 2003). The following notation is used:

3 Proposed signature schemes

In this section, we propose two signature schemes as building blocks for constructing anonymous credential system. One is the extended BLS signature scheme, the other is the extended BGLS aggregate signature scheme. The following notation is used:

- G_1 and G_2 are two (multiplicative) cyclic groups of prime order p, and (G_1, G_2) is a co-GDH group
- g_2 is a fixed generator of G_2
- ψ is an efficiently computable isomorphism from G_2 to G_1, with $\psi(g_2) = g_1$ and
- e is an efficiently computable bilinear map $e : G_1 \times G_2 \rightarrow G_T$ with bilinear and non-degenerate properties
- H is a full-domain hash function $H : \{0, 1\}^* \rightarrow G_1$

3.1 Construction of BLS+ signature scheme

We extend the BLS short signature (Boneh et al. 2001) to be able to construct anonymous credentials for weak attributes, where the user binds a private value to attributes, so as to prove the possession of credential to the verifier. The extended BLS signature scheme, called BLS+ in this paper, uses a full-domain hash function H to abstract a message $m \in \{0, 1\}^*$. The security analysis views H as a random oracle.

BLS+.KeyGen Pick random values $x \in_R Z_p$ and compute $v \leftarrow g_2^x \in G_2$. The public key is v, The private key is x.

BLS+.Sign Given a private key x, a message tuple (m, r) where $m \in \{0, 1\}^*, r \in Z_p$, compute $h \leftarrow H(m), \sigma \leftarrow (h \cdot g_1)^x$, then output $\sigma \in G_1$ as a signature.

BLS+.Verify Given a public key $v \in G_2$, a message tuple (m, r) where $m \in \{0, 1\}^*, r \in Z_p$, and a signature $\sigma \in G_1$, compute $h \leftarrow H(m)$, output $true$ if $e(\sigma, g_2) = e(h \cdot g_1, v)$ holds.

Theorem 1 *Let (G_1, G_2) be a co-GDH group pair of order p. The BLS+ signature scheme on (G_1, G_2) is existentially unforgeable against adaptively chosen-message attack only if the input messages in $\{0, 1\}^*$ are distinct.*

Suppose A is a forger algorithm that breaks the BLS+ signature scheme. We show how to construct an algorithm B breaking BLS signatures that are secure under co-CDH assumption. Algorithm B is given g_2 and v, where $v = g_2^x \in G_2$. B simulates the challenger and interacts with forger A as follows.

Setup B starts by giving A the generator g_2 and the public key v.

Signature queries Let $(m_1, r_1), \ldots, (m_{q_s}, r_{q_s})$, where $m_i \in \{0, 1\}^*, r_i \in Z_p$, be signature queries issued by A. For each i, B is given access to **BLS.Sign** to obtain the signatures on m_i as referred in Boneh et al. (2001). Next, B defines $\sigma_i \leftarrow H(m_i)^x \cdot \psi(v)^{r_i}$. Observe that $e(\sigma_i, g_2) = e(H(m_i)^x \cdot \psi(v)^{r_i}, g_2) = e(H(m_i) \cdot g_1^{r_i}, v)$ and therefore σ_i is a valid signature on (m_i, r_i) under the public key v. B gives σ_i to A.

Output Eventually we assume A produces a valid message-signature tuple (m_f, r_f, σ_f) where $m_f \notin \{m_1, ..., m_{q_s}\}$. From the verification equations, we have $e(\sigma_f, g_2) = e(H(m_f) \cdot g_1^{r_f}, v)$. It follows that $\sigma_f = (H(m_f) \cdot g_1^{r_f})^x = H(m_f)^x \cdot \psi(v)^{r_f}$. Then, B outputs $H(m_f)^x \in G_1$ as $H(m_f)^x \leftarrow \sigma_f / \psi(v)^{r_f}$. This means that a BLS signature for a new message m_f is forged, which contradicts co-CDH assumption.

Theorem 2 *Let (G_1, G_2) be a co-GDH group pair of order p. The BLS+ signature scheme on (G_1, G_2) is existentially*

unforgeable against known-message attack where the input tuple is $(m \in \{0,1\}^*, M \in G_1)$ *and the messages in* $\{0,1\}^*$ *might be the same, if and only if* $r = log_{g_1} M$ *is hidden all the way but M can be opened in the proof of knowledge, and there is no chance for any signature requester obtaining two different signatures on the same message m.*

We show A breaks co-CDH assumption on (G_1, G_2). This will contradict the fact that (G_1, G_2) is a co-GDH group pair. Assume A is not given access to any sign oracle, instead, it is given access to n known message-signature tuples $(m_1, M_1, \sigma_1), ..., (m_n, M_n, \sigma_n)$ under the public key v. Note that $M_1, ..., M_n$ are uniform in G_1 and are independent of A's view. We assume A outputs a valid message-signature tuple (m_f, M_f, σ_f), where $m_f = m_i, M_f \neq M_i$ for some $i \in \{1, ..., n\}$.

Along with the verification equation, A also has to take a proof of knowledge that it can open the commitment M_f, and extracts $r_f = log_{g_1} M_f$. Assume $e(\sigma_f, g_2) = e(H(m_f) \cdot M_f, v)$ is accepted, since $H(m_f) = H(m_i)$, we have $e(\sigma_f, g_2) = e(H(m_i) \cdot M_f, v)$. We also have $e(\sigma_i, g_2) = e(H(m_i) \cdot M_i, v)$, thus

$$
\begin{aligned}
&e(\sigma_f, g_2) = [e(\sigma_i, g_2)/e(M_i, v)]e(M_f, v)\\
&e(\sigma_f, g_2)e(M_i, v) = e(\sigma_i, g_2)e(M_f, v)\\
&\sigma_f \cdot M_i^x = \sigma_i \cdot M_f^x\\
&g_1^{(r_i - r_f)x} = \sigma_i/\sigma_f
\end{aligned}
$$

Then, A outputs $g_1^{(r_i - r_f)x}$. This contradicts co-CDH assumption if r_i is hidden from A.

3.2 Construction of BGLS+ aggregate signature scheme

We modify BGLS signature scheme to construct attributes proof protocols. The modified one, called BGLS+ signature scheme in this paper, comprises five algorithms (KeyGen, Sign, Verify, Aggregate and AggregateVerify). In particular, the algorithm Aggregate is on a subset of messages from a single message holder. Without loss of generality, the identity-like information could be distinct with each message as required. The algorithms KeyGen, Sign and Verify are identical to BLS+ signature scheme, while the algorithm Aggregate and AggregateVerify are specified as follows.

BGLS+.Aggregate For the aggregating subset of messages hold by the same user, assign to each message an index i, ranging from 1 to k. The user provides $\sigma_i \in G_1$ on each message tuple (m_i, r_i) of his choice, where $m_i \in \{0,1\}^*, r_i \in Z_p$. Compute $\sigma \leftarrow \prod_{i=1}^k \sigma_i$ for k message tuples, and output the aggregate signature $\sigma \in G_1$.

BGLS+.AggregateVerify Given an aggregate signature $\sigma \in G_1$ for an aggregating subset of messages hold by the same user, indexed as before, the original message tuple (m_i, r_i) where $m_i \in \{0,1\}^*, r_i \in Z_p$, and public key v_i, the verifier computes $h_i \leftarrow H(m_i)$; outputs $true$ if $e(\sigma, g_2) = \prod_{i=1}^k e(H(m_i) \cdot g_1^{r_i}, v_i)$ holds.

Theorem 3 *Let* (G_1, G_2) *be a bilinear group pair of order p. The bilinear BGLS+ aggregate signature scheme on* (G_1, G_2) *is secure against existential forgery under adaptive chosen-message and aggregate chosen-key model only if the input messages in* $\{0,1\}^*$ *are distinct.*

Suppose A is a forger algorithm that breaks the BGLS+ signature. We show how to construct an algorithm C breaking BLS+ signatures. Algorithm C is given g_2 and v_1, where $v_1 = g_2^x \in G_2$. C simulates the challenger and interacts with forger A as follows.

Setup C starts by giving A the generator g_2 and the public key v_1.

Signature queries A requests a signature on some message (m, r), where $m \in \{0,1\}^*, r \in Z_p$, under the challenge key v_1. First, C is given access to **BLS.Sign** to obtain the signature $H(m)^x$ on the message m under the public key v_1 as referred in Boneh et al. (2001). Next, C computes $\sigma \leftarrow H(m)^x \cdot \psi(v_1)^r$. Observe that $e(\sigma, g_2) = e(H(m)^x \cdot \psi(v_1)^r, g_2) = e(H(m) \cdot g_1^r, v)$ and therefore σ is a valid signature on (m, r) under the public key v_1. C gives σ to A.

Output Finally, A returns additional $k-1$ public keys $v_2, \ldots, v_k$, k message tuples $(m_1, r_1), \ldots, (m_k, r_k)$, and a forged aggregate signature $\sigma \in G_1$. The messages m_i must all be distinct, and A must not have requested a signature on m_1.

For each $m_i, 2 \leq i \leq k$, C first issues a query to **BLS.Sign** to obtain the BLS signatures on m_i under the public key v_i, i.e., $H(m_i)^x \leftarrow \psi(v_i)^{b_i}$ as referred in Boneh et al. (2001). Next, for each $i > 1$, C sets $\sigma_i \leftarrow H(m_i)^x \cdot \psi(v_i)^{r_i}$. Then, for $i > 1$, $e(\sigma_i, g_2) = e(H(m_i)^x \cdot \psi(v_i)^{r_i}, g_2) = e(H(m_i) \cdot g_1^{r_i}, v_i)$. So σ_i is a valid BLS+ signature on (m_i, r_i) by the key whose public component is v_i for $i > 1$.

Now, C constructs $\sigma_1 : \sigma_1 \leftarrow \sigma \cdot (\prod_{i=2}^k \sigma_i)^{-1}$. Then,

$$
\begin{aligned}
e(\sigma_1, g_2) &= e(\sigma, g_2)\left(\prod_{i=2}^{k} e(\sigma_i, g_2)\right)^{-1}\\
&= \prod_{i=1}^{k} e(H(m_i) \cdot g_1^{r_i}, v_i) \cdot \left(\prod_{i=2}^{k} e\left(H(m_i) \cdot g_1^{r_i}, v_i\right)\right)^{-1}\\
&= e(H(m_1) \cdot g_1^{r_1}, v_1).
\end{aligned}
$$

Thus, $\sigma_1 = (H(m_1) \cdot g_1^{r_1})^x$, C outputs a BLS+ signature on a new message m_1 under the private key x, which breaks BLS+ signature.

4 Construction of anonymous credentials

In this section, we describe a novel anonymous credential system, where the BLS+ signature scheme is used to issue an individual credential, and the BGLS+ signature scheme is used to prove multiple credentials all at once.

As the variant BLS+ signature scheme is applied to construct an anonymous credential system, the premise of security forces the Issuer to concern such cases as follows.

The signer is able to prevent any signature requester from having two different signatures σ_1, σ_2 on the same message m. Precisely, the signer may have a record of requester's id and message m, and do duplication check every time when receive a signature query. It will be denied of issuance when such duplication is detected.

The value r is supposed to be kept private all the way. To prevent collusion between signature requesters from forging a signature by sharing their private values, the signer will bind the same private value r with a user's weak attributes and strong ones also. Given an adversary A sharing the private value r with any colluding partner B, A will be at a highly risk to be impersonated by B in the case of showing a credential on some strong attribute.

The anonymous credential system consists of two basic protocols, i.e., issuance and showing, as well as attributes proof protocol on logical relations over attributes. To reveal the value of a weak attribute itself will not break anonymity inherently, however, anonymous credential systems for weak attributes still need to guarantee the following security and privacy requirements.

- **Unforgeability** Only the prover with ownership of the credential is accepted by the verifier in attributes proof. Unforgeability is in compliance with the fundamental security of signature scheme.
- **Untraceability** Issuers are unable to trace issued attributes and their owners. In the other word, issuance and showing of a credential are mutually unlinkable.
- **Unlinkability** Any verifier cannot determine whether any pair of attributes proof is conducted by the same user even by colluding with other verifiers.
- **Selective disclosure of attributes** Users can select which portions of a credential to reveal, which portions to keep hidden, and what relations between certified items are exposed during attribute proofs.

4.1 Attributes encoding

In general cases, an attribute implies a tuple (*id*, *attribute type*, *attribute value*). The *id* is the identifier of the credential holder. It may be real name, pseudonym, any attribute value being an identifier, or signature. Such identifiers are different for each credential holder and can be identified by the Issuer. Setting up an *id* with attributes makes credential issuance more practical, because in the physical world issuing authorities tend to identify the user before asserting his attributes and issuing him a credential. In our case, the *id* is mapped to M and constructed as $M = g_1^r$ where r is a secret value chosen by the user. However, the user does not always need to reveal his *id* when showing a credential or proving attributes as far as anonymity is concerned.

Most of attribute values are corresponding to a single attribute type in the universal attributes field. They can be pre-defined in a finite set and generally regarded as weak attributes; that means they can hardly identify the user in a population. The verifier's policy generally requires users prove either possession of all of the multiple attributes or possession of one of the multiple attributes. For example, when submitting a resume, a person has to show a credential with the multiple attributes (gender, $female$), (nationality, $French$) and (degree, $Ph.D$) all together embedded, while in the other scenario, one person can enjoy the free tickets with his ID-card only if any one of the multiple attributes (minority, $blind$), (social_benefit, $unemployed$) or (type, $kids\_card$) is embedded. For simplifying attributes proof, we assume there are not any two different attribute labels assigned with identical values. It means we can distinguish an attribute from the value. Back to the above examples, when submitting a resume, a person has to show a credential with all of the multiple attribute values $female$, $French$, $Ph.D$ embedded, while one person can enjoy the free tickets with any one of the multiple attribute values $blind$, $unemployed$, $kids\_card$ in his ID-card.

Generally, the anonymous credential is a cryptographic digital signature on attribute values. Prior to attributes encoding, we distinguish weak attributes from strong ones. The random oracle model can be utilized in encoding weak attributes since the values of them are not identifiable inherently and can be revealed in the clear without breaking anonymity. The requirement of information-theoretically secure of attribute is relaxed, so the ith attribute is encoded as the tuple $(M, H(m_i))$ in the random oracle model where $M = g_1^r$ is id, instead of encoded all attributes as a discrete logarithm representation. To solve linear complexity of attributes proof, an anonymous credential only encodes a single one attribute instead of multiple ones. It is fundamentally a BLS+ signature on a single one attribute. Thus, the signature on the attribute tuple (M, m_i) is σ_i as $\sigma_i = (H(m_i) \cdot M)^x$.

4.2 Issuance protocol

The issuance protocol is for the signer to certify the user's attributes with its signing key and release a set of BLS+ signatures, each of which is on a single attribute value. During this procedure, the User generates the commitment of a private value for each single attribute for the proof of ownership

later on. It is sufficient for the Issuer to know the commitment instead of the actual value of this private part, however, the Issuer needs to verify the form of the commitment to avoid the user's fraud. The random values will always be kept secret as showing the credential, and only the owner who knows it can prove the ownership of the credential.

Common input The public parameters $(p, G_1, G_2, G_T, e, H, g_1, g_2, v)$, where $v = g_2^x$, $H : \{0,1\}^* \rightarrow G_1$, attribute values $m_1, \ldots, m_n \in \{0,1\}^*$, and the commitment $M \in G_1$.

User's input Value $r \in_R Z_p$ such that $M = g_1^r$.

Issuer's input Signing key x.

1. The Issuer gives a zero-knowledge proof of knowledge with the User:

   $$PK\{\gamma : M = g_1^{\gamma}\}$$

2. For each $m_i, 1 \leq i \leq n$, the Issuer computes $h_i \leftarrow H(m_i), \sigma_i \leftarrow (h_i \cdot M)^x$. Thus, σ_i is the credential of the attribute m_i.
3. The Issuer sends $\sigma_1, \ldots, \sigma_n$ to the User.

Theorem 4 *The issuance protocol is a secure two-party computation of a signature on a discrete logarithm representation of g_1^r under the signer's public key.*

From the signer's point of view, this protocol is as secure as when the user submits his signature queries in the clear. This is because of the proof of knowledge: there exists an extractor that can discover the value of the message being signed, and ask it to the signer in the clear.

From the user's point of view, since the user's secret input r is only used in the zero-knowledge proof of knowledge of it, the only thing that the signer finds out about the value r is the input value $M = g_1^r$. The hardness of discrete logarithm problem makes $r = \log_{g_1} M$ unknown.

5 Attributes proof

In this section, we present two selective disclosure-enabled attributes proof protocols constructed by the proposed anonymous credential. The user can prove (1) the possession of all of the multiple attributes, i.e., AND relation over attributes. (2) the possession of one of multiple attributes, i.e., OR relation over attributes.

We adopt BGLS+ aggregate signature scheme to aggregate any combination of credentials into a single one. Particularly, we consider the case when all individual credentials are issued by the single Issuer under the public key v, that is the same case to any conventional credential which embeds all the attributes certified by the same Issuer together.

5.1 AND relation proof

For AND relation, it is to prove that a specified set of attributes $\{a_1, \ldots, a_L\}$ are all certified. We define two sets for attributes proof, $ATTR$ and TA. $ATTR$ is the set of all the values of user's attributes certified by the Issuer, i.e., $ATTR = \{m_1, \ldots, m_N\}$. It is encoded in a credential $Cred$ which is formed as $Cred = (M, ATTR, \{\sigma_1, \ldots, \sigma_N\})$, where M is the value of Prover's id formed as $M = g_1^r$. TA is made up of the values of the attributes referenced in a proof, i.e., $TA = \{a_1, \ldots, a_L\}, 1 \leq L \leq N$. It is specified by the Verifier's policy. The Prover may show some or all of the attributes, however, the actual value of id remains private at all time. Only the Prover who can prove the knowledge of secret value r is truly the owner of credential. Therefore, the credential constructed this way is against identity theft. The protocol is a zero-knowledge proof of knowledge of a BGLS+ aggregate signature on multiple attributes.

Pre-computation Given the Issuer's public parameters $(p, G_1, G_2, G_T, e, H, g_1, g_2, v)$ where hash function $H : \{0,1\}^* \rightarrow G_1$, the Verifier pre-computes $h \leftarrow \prod_{i=1}^{L} H(a_i)$ according to $TA = \{a_1, \ldots, a_L\}$, $V_m \leftarrow e(h, v)$ and $V_c \leftarrow e(g_1, v)$. The Prover pre-computes $V_c \leftarrow e(g_1, v)$.

Common input: The public parameter $(p, G_1, G_2, G_T, e, H, g_1, g_2, v)$, hash function $H : \{0,1\}^* \rightarrow G_1$, $TA = \{a_1, \ldots, a_L\}$, V_c.

Verifier's input: V_m.
Prover's input: Value r such that $M = g_1^r$, $ATTR = \{m_1, \ldots, m_N\}, \sigma_1, \ldots, \sigma_N$.

1. The Prover aggregates the signatures of the proved attributes referenced in TA as follows:

   $$h \leftarrow \prod_{i=1}^{L} H(a_i)$$

   $$V_m \leftarrow e(h, v)$$

   $$\sigma \leftarrow \prod_{1 \leq i \leq N}^{m_i \in TA \cap ATTR} \sigma_i$$

   $$r \leftarrow r \cdot L$$

2. The Prover computes a blinded version of the aggregated signature σ: Choose a random value $r' \in_R Z_p$, and blind the signature to form $\sigma' \leftarrow \sigma^{r'}$, where σ' is distributed independently of everything else. Then, the Prover sends σ' to the Verifier.

3. Let $V_s = e(\sigma', g_2)$, the Prover and Verifier carry out the following zero-knowledge proof protocol:

   $PK\{(\alpha, \beta) : {V_s}^{\alpha} = V_m {V_c}^{\beta}\}$

   The Verifier accepts if the proof above is correct.

Theorem 5 *The showing protocol is a zero-knowledge proof of knowledge of a BGLS+ signature on multiple distinct message tuples.*

First, we prove the zero-knowledge property. The values that the Verifier receives from the Prover in Step 2 are independent of the actual signature: σ' is random in G_1 because $\sigma' = \sigma^{r'}$ for a random chosen r'. Therefore, consider the following simulator S: Choose random r', and set $\sigma' = g_1^{r'}$. Then, σ' is distributed correctly, and so Step 2 is simulated correctly. Then, since in Step 3, the Prover and Verifier execute a zero-knowledge proof, it follows that there exists a simulator S' for this step; just run S'. It shows that S constructed this way is a zero-knowledge simulator for this protocol.

Next, we prove the proof of knowledge property. A knowledge extractor E is exhibited to output values $(a_1, \ldots, a_L, r, r', \sigma)$, such that σ is a valid aggregate signature on $(a_1, g_1^r), \ldots, (a_L, g_1^r)$. The extractor E is given access to the Prover such that the Verifier's acceptance probability is non-negligible. It proceeds as follows: first, it runs for the proof of knowledge protocol of Step 3. As a result, it obtains the values $r, r' \in Z_p$ such that ${V_s}^{r'} = V_m {V_c}^{r}$.

We wish to show that $(a_1, \ldots, a_L, g_1^r)$ and $\sigma = \sigma'^{r'}$ satisfy the verification equation for BGLS+ under the public key pair v: $e(\sigma'^{r'}, g_2) = \prod_{i=1}^{L} e(h_i g_1^r, v)$. We have:

$${V_s}^{r'} = V_m {V_c}^{r}$$

$$e(\sigma', g_2)^{r'} = e(h, v) e(g_1, v)^{r}$$

$$e(\sigma'^{r'}, g_2) = e\left(\prod_{i=1}^{L} h_i g_1^r, v\right)$$

$$e(\sigma'^{r'}, g_2) = \prod_{i=1}^{L} e(h_i g_1^r, v)$$

5.2 OR relation proof

The Prover needs to prove that one of the subset of attributes is signed in the credential. Given L items of elementary predicates, i.e., $\exists m_j | (m_j = a_1) \vee (m_j = a_2) \vee \cdots \vee (m_j = a_L)$, $j \in \{1, \ldots, N\}$, the OR relation proof implies to prove one of the attributes in TA, where $TA = \{a_1, \ldots, a_L\}$, is embedded into the user's credential. For privacy concern, the Verifier cannot distinguish which the particular one in TA is. It is required that the proof of the particular attribute m_j is hidden in the proof of all L values $a_1, \ldots, a_L$. In the zero-knowledge proof of a signature on L values, only the value of the particular one is actually signed while others are redundant to protect it from being distinguished.

Pre-computation Given the Issuer's public parameters $(p, G_1, G_2, G_T, e, H, g_1, g_2, v)$ where hash function $H : \{0, 1\}^* \rightarrow G_1$, the Verifier pre-computes $h_i \leftarrow H(a_i)$, $V_i \leftarrow e(h_i, v)$ for each a_i in $TA = \{a_1, \ldots, a_L\}$ and $V_c \leftarrow e(g_1, v)$. The Prover pre-computes $\hat{h}_i \leftarrow H(m_i)$, $\hat{V}_i \leftarrow e(\hat{h}_i, v)$ for each m_i in $ATTR = \{m_1, \ldots, m_N\}$, and $V_c \leftarrow e(g_1, v)$.

Common input: The public parameter $(p, G_1, G_2, G_T, e, H, g_1, g_2, v)$ where hash function $H : \{0, 1\}^* \rightarrow G_1$, $TA = \{a_1, \ldots, a_L\}$, V_c.

Verifier's input: $V_1, \ldots, V_L$.

Prover's input: Value r such that $M = g_1^r$, $ATTR = \{m_1, \ldots, m_N\}, \sigma_1, \ldots, \sigma_N, \hat{V}_1, \ldots, \hat{V}_N$.

1. The Prover picks the required attributes according to TA and generates a subset $\{V_1, \ldots, V_L\}$ from $\{\hat{V}_1, \ldots, \hat{V}_N\}$.
2. Suppose the Prover wants to prove $m_j = a_k$ where $j \in \{1, \ldots, N\}, k \in \{1, \ldots, L\}$. It chooses a random value $r' \in_R Z_p$ and generates a blinded version of the corresponding signature, $\sigma' \leftarrow {\sigma_j}^{r'}$, then sends σ' to the Verifier.
3. Let $V_s = e(\sigma', g_2)$, the Prover and Verifier carry out the following zero-knowledge proof of knowledge:

   $PK\{(\alpha_1, \alpha_2, \ldots, \alpha_L, \beta) : V_s = V_1^{\alpha_1} V_2^{\alpha_2} \ldots V_L^{\alpha_L} {V_c}^{\beta}\}$

   The Verifier accepts if the proof above is correct.

Theorem 6 *The OR relation proof protocol is a zero-knowledge proof of knowledge of a BLS+ signature.*

First, we prove the zero-knowledge property. The values that the Verifier receives from the Prover in Step 2 are independent of the actual signature: σ' is random in G_1 because $\sigma' = \sigma^{r'}$ for a random chosen r'. Therefore, consider the following simulator S: Choose random r',and set $\sigma' = g_1^{r'}$. Then, σ' is distributed correctly, and so Step 2 is simulated correctly. Then, since in Step 3, the Prover and Verifier execute a zero-knowledge proof, it follows that there exists a simulator S' for this step; just run S'. It shows that S constructed this way is a zero-knowledge simulator for this protocol.

Next, we prove the proof of knowledge property. We must exhibit a knowledge extractor E that, given access to a Prover such that the Verifier's acceptance probability is non-negligible, outputs $(a_1, \ldots, a_L, w_1, \ldots, w_L, r, \sigma)$, such that σ is a valid signature on $(a_1, g_1^r), \ldots, (a_L, g_1^r)$. Given such a Prover, the extractor proceeds as follows: first, it

runs for the proof of knowledge protocol of Step 3. As a result, it obtains the values $w_1, \ldots, w_L, r \in Z_p$ such that $V_s = V_1^{w_1} V_2^{w_2} \ldots V_L^{w_L} V_c{}^r$.

Case I [$\exists w_j \neq 0, j \in \{1, \ldots, L\}, \forall w_i = 0, i \in \{1, \ldots, L\}, i \neq j$]: we wish to show that $(a_j, g_1^{r/w_j})$ and $\sigma = \sigma'^{1/w_j}$ satisfy the verification equation for the BLS+ signature scheme: $e(\sigma'^{1/w_j}, g_2) = e(h_j g_1^{r/w_j}, v)$ where $h_j = H(a_j)$. We have:

$$V_s = V_1^{w_1} V_2^{w_2} \ldots V_L^{w_L} V_c{}^r$$

$$e(\sigma', g_2) = e(h_j, v)^{w_j} e(g_1, v)^r$$

$$e(\sigma'^{1/w_j}, g_2) = e(h_j g_1^{r/w_j}, v)$$

Case II [$\exists \{w_j = w \neq 0\}, j \in \{1, \ldots, L\}, \forall w_i = 0, i \in \{1, \ldots, L\}, i \notin \{j\}$]: we wish to show that $(\{a_j\}, g_1^{r/w})$ and $\sigma = \sigma'^{1/w}$ satisfy the verification equation for the BGLS+ signature scheme: $e(\sigma'^{1/w}, g_2) = \prod_{1 \leq j \leq L}^{w_j = w} e(h_j g_1^{r/w}, v)$, where $h_j = H(a_j)$ for each j. We have:

$$V_s = V_1^{w_1} V_2^{w_2} \ldots V_L^{w_L} V_c{}^r$$

$$e(\sigma', g_2) = \prod_{1 \leq j \leq L}^{w_j = w} e(h_j, v)^w e(g_1, v)^r$$

$$e(\sigma'^{1/w}, g_2) = \prod_{1 \leq j \leq L}^{w_j = w} e(h_j g_1^{r/w}, v)$$

$$e(\sigma'^{1/w}, g_2) = \prod_{1 \leq j \leq L}^{w_j = w} e(h_j, v) e(g_1^{r_j/w}, u),$$

$$\text{where } \sum_{j=1}^{L} r_j = r$$

Case III [$\exists w_j \neq w \neq 0, j \in \{1, \ldots, L\}, \forall w_i = w, i \in \{1, \ldots, L\}, i \neq j$]: Let $r \leftarrow r \cdot L$, we wish to show that this case is negligible. We have:

$$V_s = V_1^{w_1} V_2^{w_2} \ldots V_L^{w_L} V_c{}^r$$

$$e(\sigma', g_2) = \prod_{1 \leq i \leq L}^{i \neq j} e(h_i{}^w, v) e(h_j{}^{w_j}, v) e(g_1^r, v)$$

$$e(\sigma'^{1/w}, g_2) = \prod_{1 \leq i \leq L}^{i \neq j} e(h_i h_j{}^{w_j/w}, v) e(g_1^{r/w}, v)$$

$$e(\sigma'^{1/w}, g_2) = \prod_{1 \leq i \leq L}^{i \neq j} e(H(a_i) H(a_j)^{w_j/w}, v) e(g_1^{r/w}, v)$$

Thus, $\sigma'^{1/w} = \prod_{1 \leq i \leq L}^{i \neq j} H(a_i)^x (H(a_j)^{w_j/w})^x (g_1^{r/w})^x$, then, $\sigma'^{1/w}$ is the valid BGLS+ signature on $\{(a_i, g_1^{r_i/w}) | 1 \leq i \leq L, i \neq j\}$ and $(a^*, g_1^{r_i/w})$, where $r_i = r/L$ and $H(a^*) = H(a_j)^{w_j/w}$. It contradicts the collision resistance property of hash function.

6 Efficiency

In this section, we compare the efficiency of AND and OR relation proof with the conventional Camenisch–Lysyanskaya pairing-based system (short for CL system) (Jan and Anna 2004), accumulator-based system (Amang et al. 2011) and our aggregation-based anonymous credential system. The computational complexity with respect to the number of exponentiations and pairings is mainly considered, while the multiplication and hash function is omitted, since the costs of them are much smaller. The following parameters are used.

N: the total number of the attributes certified by the Issuer.

L: the number of the attributes referenced in a proof.

N_S: the total number of the string attributes out of N.

L_S: the number of the string attributes, out of L, referenced in a proof.

The bilinear maps used in the CL system and accumulator-based system is $e : G_1 \times G_1 \rightarrow G_T$, while in our aggregation-based system is $e : G_1 \times G_2 \rightarrow G_T$.

$E(G_1)$: exponentiations on G_1.

$E(G_T)$: exponentiations on G_T.

Table 1 shows the total number of exponentiations in AND relation proof with different systems. It is the addition of *randomization of a signature*, *generation of a proof* and *verification of a proof*. The first two parts are related to the Prover, while the last one is related to the Verifier. In our system, the number of exponentiations in AND relation proof is constant with the number of attributes, whether they are string or finite-set.

Table 2 shows the total number of pairings in AND relation proof with different systems. It is the addition of *generation a proof* and *verification of a proof*. In our system, the number of pairings in AND relation proof is 3 and outperforms the accumulator-based system with constant complexity.

Table 3 shows the total number of exponentiations in OR relation proof between the accumulator-based system and our aggregation-based system (there is no OR relation proof presented in the conventional CL system). It is the addition of *randomization of a proof*, *generation of a proof* and *verification of a proof*, as the same as AND relation proof. In the accumulator-based system, encoding finite-set attribute values does not need any exponentiation due to accumulator, while encoding a string attribute on the base also needs to be computed by the Verifier every single time. Both the accumulator-based system and our aggregation-based

Table 1 Number of exponentiations in AND relation proof

	CL system	Accumulator-based system	Aggregation-based system
Prover	$(4+2N)E(G_1)+(2+N-L)E(G_T)$	$24E(G_1)+(N_S+15)E(G_T)$	$E(G_1)+2E(G_T)$
Verifier	$LE(G_1)+(N-L+3)E(G_T)$	$(N_S+20)E(T)+15E(G)$	$3E(G_T)$

Table 2 Number of pairings in AND relation proof

	CL system	Accumulator-based system	Aggregation-based system
Prover	$2+N-L$	4	2
Verifier	$6+5N-L$	13	1

Table 3 Number of exponentiations in OR relation proof

	Accumulator-based system	Aggregation-based system
Prover	$47E(G_1)+(N_S+26)E(G_T)$	$E(G_1)+(L+1)E(G_T)$
Verifier	$(N_S+69)E(G_T)$	$(L+2)E(G_T)$

Table 4 Number of pairings in OR relation proof

	Accumulator-based system	Aggregation-based system
Prover	8	1
Verifier	23	1

system have linear complexity with respect to exponentiation in OR relation proof, however, the aggregation-based one costs less than the accumulator-based one as long as $E(G_1)+(2L+3)E(G_T) \leq 47E(G_1)+(2N_S+95)E(G_T)$. Note that the finite-set attributes dominate the attribute types and the string attributes take up relatively smaller portion; besides, the exponentiation cost on G_T is larger than that on G_1. Using an example of eID as in Jan and Thomas (2008) where $N_S \leq 5$, $N \leq 45$ (thus $L \leq 45$), the aggregation-based system outperforms the accumulator-based system.

Table 4 shows the total number of pairings in OR relation proof between the accumulator-based system and our aggregation-based system. Pre-computation removes the pairings which are irrelevant to the randomized signatures. In our system, the number of pairings in OR relation proof is 2 and outperforms the accumulator-based system with constant complexity.

Regarding the public parameters to construct an anonymous credential embedding N attributes, $(p, G_1, G_T, e, g_1, g_T, X, Y, \{Z_i\}, \{W_i\})$ where $1 \leq i \leq N$ are generated in the CL system, $(p, G_1, G_T, e, g, \tilde{g}, \hat{g}, \{g_i\}, \{h_j\}, \{\tilde{S}_k, \tilde{T}_k, \tilde{U}_k, \tilde{F}_k\}, z, Y, \tilde{Y}, \hat{Y}, \tilde{Y}', \hat{Y}')$, $1 \leq i \leq 2N_F, i \neq N_F+1, 1 \leq j \leq N_S+1, 1 \leq k \leq N_F$, where N_F is the total number of finite-set attribute types, are generated in the accumulator-based system, while $(p, G_1, G_2, G_T, e, H, g_1, g_2, v)$ are generated in our aggregation-based system. We can see that the length of the public parameters in our system becomes constant.

Regarding the length of signatures with respect to the number of attributes n, there are totally $2n+3$ signatures in the CL system, formed as $(a, \{A_i\}, b, \{B_i\}, c)$. There are totally 8 signatures in the accumulator-based system, formed as (A, S, T, U, F, x, w, r). There are totally n signatures in our aggregate-based system, formed as $\{\sigma_i\}$. It turns out the aggregate-based system costs linear complexity for issuing and storing the signatures. However, when proving multiple attributes, there are only one aggregate signature involved. Therefore, our system has the shortest length of signatures in the context of attributes proof.

7 Related work

Anonymous Credential can provide strong authentication, minimal information/data disclosure, and ensure correctness of the data revealed. It was proposed by Chaum (1985) and fully implemented by Brands (Stefan 1993) and Camenisch and Lysyanskaya (Jan and Anna 2004). Brands (Stefan 1993) made use of blind signatures. Such constructions are secure in the random oracle model and very practical. Unfortunately, the resulting credentials are one-show. They are implemented by Microsoft in their U-Prove technology. Camenisch and Lysyanskaya (Jan and Anna 2004) made use of group signatures. The resulting anonymous credential systems (short for the CL system) are less efficient than the Brands', but they are multi-show since they use the inherent unlinkability property of group signatures. This technology is implemented by IBM for their Idemix product.

In recent years, cryptography based on bilinear mappings has greatly progressed. The structure of signature can be simplified by the properties of bilinear mappings. A lot of pairing-based anonymous credential systems have been proposed. Belenkiy et al. (2008); Malika et al. (2011); Nguyen and Safavi-Naini (2005) presented non-interactive anonymous credentials which can be used for non-interactive zero-knowledge proof on bilinear mapping groups. Norio et al. (2008) presented an efficient anonymous credential system which provides anonymity, unlinkability and com-

putational unforgeability under the strong Diffie–Hellman assumption. An anonymous credential is fundamentally a digital signature. Many signature schemes have been put forward to construct it, such as BB signature in Jan et al. (2009), CL signature in Jan and Anna (2004), and variant BB signatures in Amang et al. (2011). Similar to our approach, Abhilasha et al. (2010) used BGLS aggregate signature scheme to prove multiple credentials all at once for identity verification, however, their work is for the real name authentication instead of anonymous authentication. Sébastien and Roch (2011) used indexed aggregate signature to the anonymous credential system which efficiently enables a user to prove the possession, in an untraceable way, of several credentials issued by possibly several organizations. However, they did not focus particularly on attributes proof like in our work.

Strong authentication and according authorization based on certified attributes is paramount for protecting critical information and infrastructures online. Camenisch et al. (2013) propose privacy-preserving attribute-based credentials (Privacy-ABCs) in authentication and authorization systems. Our work can provide an efficient solution in its language framework to construct an anonymous credential system and related attributes proof protocols. Concerned about attributes proof, Li and Li (2006a, b) proposed oblivious attribute certificates (OACerts) and oblivious commitment-based envelope (OCBE), the user obtains a service if and only if the attribute values satisfy the service provider's policy, yet the service provider learns nothing about the actual value of attribute. Abhilasha et al. (2010) presented the multi-factor identity attributes verification scheme with hidden commitments. Yan and Dengguo (2012) used anonymous credential to propose an anonymous credential with constant complexity attribute proof. Compared with other constant complexity pairing-based systems, our system can support more types of attribute relations while the public parameter is much shorter.

Other works related to minimal information disclosure, which are also our privacy requirement, are as follows. David et al. (2008) proposed using a Merkle hash tree structure, whereby it is possible for a single certificate to contain many separate claims or attributes, each of which may be proved independently, without revealing the others. Federica et al. (2009) supported selective and incremental disclosure of identity attributes, while minimal credential disclosure guarantees that only the attributes necessary to complete the online interactions are disclosed. Patrik et al. (2011) proposed an adequate claim language specifying which certified data a user wants to reveal to satisfy a policy and provides translation algorithms for generating the anonymous credentials providing the data to be revealed.

Efficiency considerations are important when using anonymous credentials for attributes proof. The CL system suffers from the linear complexity in the total number of attributes. The state-of-the-art of attributes proof mainly focuses on reducing the number of exponentiations related to binary or finite-set attributes. Jan and Thomas (2008) extended the CL system on the strong RSA assumption for boosting the efficiency. It compresses binary and finite-set attributes into a single attribute base. The core idea is to encode discrete binary and finite-set values as prime numbers, and use the divisibility property for efficient proofs of their presence or absence. Jan et al. (2009) and Man et al. (2009) adopted the cryptographic accumulator to solve the linear complexity. Amang et al. (2011) extended the accumulator to prove AND and OR relation with constant complexity in the number of finite-set attributes. To the best of our knowledge, although they can remove exponentiations related to finite-set attributes proof, the complexity with respect to exponentiations still depends on the number of string attributes; while the number of pairings has to be largely increased to verify the validity of accumulator, and the size of public parameters is very long; so they are still far away from being practical in resource-limited environment.

8 Conclusion

We construct a novel pairing-based anonymous credential for weak attributes in the random oracle model, the efficiency of which is improved by utilizing the concept of *aggregation*. The proposed anonymous credential only encodes a single attribute, instead of multiple ones like in the existing systems. It allows the prover to prove any combination of attributes in one round by aggregating the corresponding credentials into a single proof. As cryptographic building blocks, BLS+ signature scheme is presented to construct an individual credential, and BGLS+ signature scheme is presented to prove multiple credentials all in one round. The result of efficiency analysis shows that, the number of exponentiations and pairings in AND relation proof is constant with the number of attributes, no matter whether they are string attributes or finite-set attributes. On the other hand, although our aggregation-based system has linear complexity with respect to exponentiation in OR relation proof, it outperforms the accumulator-based system in the general case of eID, where the total number of certified attributes amounts to about 45. Additionally, the number of pairings in OR relation proof is kept only once with the number of attributes, and outperforms the accumulator-based system which is also with constant complexity. Furthermore, the signature and public parameters in our system are shorter.

Acknowledgments This work was supported by the China Natural Science Foundation [Grant Number 61402095] and Fundamental Research Funds for the Central Universities [Grant Number N120404010].

References

Akagi N, Manabe Y, Okamoto T (2008) An efficient anonymous credential system. In: LNCS, financial cryptography and data security, vol 5143, pp 272–286

Au MH, Tsang PP, Susilo W, Mu Y (2009) Dynamic universal accumulators for DDH groups and their application to attribute-based anonymous credential systems. In: LNCS, topics in cryptology C CT-RSA, vol 5473, pp 295–308

Baldimtsi F, Lysyanskaya A (2012) Anonymous credentials light. In: IACR cryptology ePrint Archive, p 298

Bauer D, Blough DM, Cash D (2008) Minimal information disclosure with efficiently verifiable credentials. In: ACM, DIM '08 proceedings of the 4th ACM workshop on digital identity management, pp. 15–24

Belenkiy M, Chase M, Kohlweiss M, Lysyanskaya A (2008) P-signatures and noninteractive anonymous credentials. In: LNCS, theory of cryptography, vol 4948, pp 356–374

Bhargav-Spantzel A, Squicciarini AC, Xue R, Bertino E (2010) Multifactor identity verification using aggregated proof of knowledge. IEEE Trans Syst Man Cybern 40:372–383

Bhargav-Spantzel A, Squicciarini AC, Xue R, Bertino E (2006) Practical identity theft prevention using aggregated proof of knowledge. In: CERIAS TR, technical report, CS Department

Bichsel P, Camenisch J, Groß T, Shoup V (2009) Anonymous credentials on a standard java card. In: ACM, CCS '09 proceedings of the 16th ACM conference on computer and communications security, pp 600–610

Bichsel P, Camenisch J, Preiss F-S (2011) A comprehensive framework enabling data-minimizing authentication. In: ACM, CCS '11 proceedings of the 18th ACM conference on computer and communications security, pp 733–736

Boneh D, Gentry C, Lynn B, Shacham H (2003) Aggregate and verifiably encrypted signatures from bilinear maps. In: EUROCRYPT 2003, LNCS, vol 2656, pp 416–432

Boneh D, Lynn B, Shacham H (2001) Short signatures from the Weil pairing. In: Asiacrypt 2001, LNCS, vol 2248, pp 514–532

Boyen X, Waters B (2007) Full-domain subgroup hiding and constant-size group signatures. In: LNCS, public key cryptography, vol 4450, pp 1–15

Brands S (1993) Untraceable online cash in wallets with observers. In: CRYPTO '93: 13th annual international cryptology conference. Springer, New York, pp 302–318

Camenisch J, Zurich Zurich, Lehmann A, Neven G (2012) Electronic identities need private credentials. IEEE Secur Priv 10:80–83

Camenisch J, Dubovitskaya M et al (2013) Concepts and languages for privacy-preserving attribute-based authentication. In: Policies and research in identity management, vol 396. Springer, Heidelberg, pp 34–52

Camenisch J, Groß T (2008) Efficient attributes for anonymous credentials. In: ACM, CCS '08 proceedings of the 15th ACM conference on computer and communications security, pp 345–356

Camenisch J, Kohlweiss M, Soriente C (2009) An accumulator based on bilinear maps and efficient revocation for anonymous credentials. In: LNCS, public key cryptography vol 5443, pp 481–500

Camenisch J, Lysyanskaya A (2004) Signature schemes and anonymous credentials from bilinear maps. In: Advances in cryptology—LNCS, vol 72. Springer, Heidelberg, pp 3152:56

Canard S, Lescuyer R (2011) Anonymous credentials from (indexed) aggregate signatures. In: ACM, DIM '11 proceedings of the 7th ACM workshop on digital identity management, pp 53–62

Canard S, Lescuyer R, Traoré J (2011) Multi-show anonymous credentials with encrypted attributes in the standard model. In: cryptology and network security, LNCS, vol 3531, pp 318–333

Chaum D (1985) Transaction systems to make big brother obsolete. In: ACM, communications of the ACM, vol28, pp 1030–1044

Guo N, Cheng J, Zhang B, Yim K (2013) Aggregate signature-based efficient attributes proof with pairing-based anonymous credential. In: 16th international conference on network-based information systems, pp 276–281

Guo N, Jin Y, Yim K (2014) Anonymous credential-based privacy-preserving identity verification for business processes. In: 2014 Eighth international conference on innovative mobile and internet services in ubiquitous computing, pp 554–559

Izabachène M, Libert B, Vergnaud D (2011) Block-wise P-signatures and non-interactive anonymous credentials with efficient attributes. In: LNCS, cryptography and coding, vol 7089, pp 431–450

Li Jiangtao, Ninghui Li (2006) OACerts: oblivious attribute certificates. IEEE Trans Dependable Secur Comput 13:340–352

Li J, Li N (2006) A Construction for general and efficient oblivious commitment based envelope protocols. In: LNCS, information and communications security, vol 4307, pp 122–138

Nguyen L, Safavi-Naini R (2005) Dynamic k-times anonymous authentication. In: LNCS, applied cryptography and network security, vol 3531, pp 318–333

Paci F, Bauer D, Bertino E, Blough DM, Squicciarini AC (2009) Minimal credential disclosure in trust negotiations. Identity Inf Soc 2:221–239

Steuer K Jr, Fernando R, Bertino E (2010) Privacy preserving identity attribute verification in windows cardspace. In: ACM, DIM '10, proceedings of the 6th ACM workshop on digital identity management, pp 13–16

Sudarsono A, Nakanishi T, Funabiki N (2011) Efficient proofs of attributes in pairing-based anonymous credential system. In: LNCS, privacy enhancing technologies, vol 6794, pp 246–263

Zhang Y, Feng D (2012) Efficient attribute proofs in anonymous credential using attribute-based cryptography. In: LNCS, information and communications security, vol 7618, pp 408–415