Efficient Attribute Proofs in Anonymous Credential Using Attribute-based Cryptography*

Yan Zhang and Dengguo Feng

Institute of Software, Chinese Academy of Sciences, Beijing, China
{janian,feng}@is.iscas.ac.cn

Abstract. As an important property of anonymous credential, attribute proof allows user to prove the possession of attributes issued by the issuing authority anonymously. In this paper, we introduced the notation of Attribute-based signature into anonymous credential to propose an anonymous credential with constant complexity attribute proof. Compared with other constant complexity pairing based schemes, our scheme could support more types of attribute relations while the public parameter is much shorter.

Keywords: Attribute-based, anonymous credential, attribute proof, efficient.

1 Introduction

Along with the widely applied electronic identification and requirements of user privacy, anonymous credential has become a research focus of authentication technology these days. Besides the basic anonymity, attribute proof plays an important role in anonymous credential systems, too. By using anonymous attribute proof, user could make an attestation to the verifier that certain attributes were issued to his credential without disclosing his identity.

In current anonymous credentials, the attribute proof usually use the proof of knowledge algorithms, however, the computational complexity and length of attestation of these schemes were linearly related to the number of attributes contained in the proof. To solve this problem, Camenisch and Gross proposed an efficient coding method and extended the CL anonymous credential with it to significantly improve the efficiency of anonymous attribute proofs in 2008 [4]. In 2011, Amang Sudarsono et al. proposed a similar scheme in Pairing-Based anonymous credentials [5].

By using relevant technologies like accumulators, both two schemes mentioned above can achieve constant complexity of finite attributes(attribute values select from a small sized finite-set, for example: gender, nationality, age etc.) proofs,

* Supported by the National Natural Science Foundation of China under Grant No. 91118006; The National High-Tech Research and Development Plan of China under Grant Nos. 2011AA01A203, 2012AA01A403; The Opening Project of Key Lab of Information Network Security of Ministry of Public Security (The Third Research Institute of Ministry of Public Security) under Grant No. C11604.

T.W. Chim and T.H. Yuen (Eds.): ICICS 2012, LNCS 7618, pp. 408–415, 2012.

which make the attribute attestation significantly efficient. Unfortunately, although the two schemes could improve the efficiency of proof, there are still some drawbacks. In the first scheme, the computational complexity was linearly related to the total number of the finite attributes. The scheme of [5] could achieve computational complexity independent with attributes' number, but the size of its public parameters was very long so it is hard to apply in resource limited environment.

Attribute-based Signatures(ABS) [1][2], which is an extension of Identity-based signature proposed by Maji et al. in 2008 [1], gives us a new idea to build efficient attributes proofs in anonymous credential. In ABS, the user's secret key contains some attribute information, which makes the signature be verified to be generated by a user holds certain attributes, while hiding the identity of the true generator. For user-anonymity and attribute attestation were already contained in the signature scheme, it can be extended into anonymous credentials with efficient attribute proofs. Furthermore, by using ABS, the attribute proof scheme could support some complex attribute-based policy which is hard to realize in common anonymous credentials. Although ABS could only support binary attribute values, we can transform the finite attributes into multiple binary attributes to solve this problem.

1.1 Our Contributions

In this paper, we propose an anonymous credential system with constant complexity attribute proof using attribute-based signature. Our main idea is to use an ABS secret key as an attribute-based token and bind string attributes with it. The proof of finite-attributes could be extracted with the sign protocol of ABS schemes and we can use knowledge proofs to prove the string attributes.

Compared with scheme [4] and [5], our scheme has following advantages: First, by using ABS, our scheme can support proofs of threshold relation, which can be extend to general predicates, which makes our scheme more flexible to using in the attribute-based access control system. Secondly, for threshold predicates, the computational complexity is independent with the total number of the possible finite attributes, which is an advantage to [4]. Finally, for the problem of oversized public parameters in [5], our scheme significantly reduces the size of user data, which is about only 1/30 to 1/300 of scheme [5].

2 Preliminaries

2.1 Bilinear Pairings

First, we review the notion of bilinear parings, let $\mathbb{G}$ and $\mathbb{G}_T$ be cyclic groups of the prime order p, where g is a generator of $\mathbb{G}$.

If there exists a mapping $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ with following properties, then we call e a bilinear pairing.

Bilinearity: $e(g^a, h^b) = e(g, h)^{ab}$ for all $g, h \in \mathbb{G}, a, b \in \mathbb{Z}_p$;

Non-dengeneracy: There exist $g \in \mathbb{G}$ such that $e(g, g) \neq 1$, in other words, the map does not send all pairs in $\mathbb{G} \times \mathbb{G}$ to the identity in $\mathbb{G}_T$.

Computability: There is an efficient algorithm to compute $e(g_1, g_2)$ for all $g_1, g_2 \in \mathbb{G}$.

2.2 Assumptions

Our scheme is based on a new assumption we called q-HPDH, the proof of its security under generic group can be found in our full paper:

Definition 1 (q-HPDH). In a prime order group $\mathbb{G}$ with order p, the q-Hidden-Polynomial-Diffie-Hellman (q-HPDH) problem is, given a tuple $(g, g^x, g^{x^2}, \ldots, g^{x^q})$ where $x \in \mathbb{Z}_p, g \in \mathbb{G}$ and distinct $(c_1, c_2, \ldots, c_{q+1}) \in \mathbb{Z}_p$, to compute a tuple $g^r, g^{r \cdot \prod_{i=1}^{q+1}(x+c_i)}$ for some hidden value $r \neq 0$.

Moreover, the following assumptions are used in our anonymous credential scheme:

Definition 2 (q-SDH [7]). In a prime order group $\mathbb{G}$ with order p, the q-Strong-Diffie-Hellman (q-SDH) problem is, given a tuple $(g, g^x, g^{x^2}, \ldots, g^{x^q})$ where $x \in \mathbb{Z}_p, g \in \mathbb{G}$, to compute $c, g^{1/(x+c)}$.

Definition 3 (q-HSDH [9]). In a prime order group $\mathbb{G}$ with order p, the q-Hidden-Diffie-Hellman Exponent(q-HSDH) problem is, given a tuple $(g, h, g^x,$ $(g^{1/(x+b_1)}, u^{b_1}, v^{b_1}), \ldots, (g^{1/(x+b_q)}, u^{b_q}, v^{b_q}))$ where $x \in \mathbb{Z}_p, g, h \in \mathbb{G}$, to compute $(g^{1/(x+b)}, u^b, v^b)$ for some b distinct from $b_i (i = 1, \ldots, q)$.

Definition 4 (q-TDH [8]). In a prime order group $\mathbb{G}$ with order p, the q-Triple-Diffie-Hellman Exponent(q-TDH) problem is, given a tuple $(g, g^x, g^y, (c_1, g^{1/(x+c_1)}), \ldots, (c_q, g^{1/(x+c_q)}))$ where $x, y \in \mathbb{Z}_p, g \in \mathbb{G}$, to compute g^{rx}, g^{ry}, g^{rxy} for some r.

2.3 BBS+ Signature

In this paper, we adopt the BBS+ signature proposed in [7] to issue the string attributes for user. This scheme is proposed as following:

Setup. Select bilinear groups $\mathbb{G}, \mathbb{G}_T$ with prime order p and a bilinear map e. Randomly select $g, g_0, h_1, \ldots, h_L \in \mathbb{G}$.

KeyGen. Select $x \in \mathbb{Z}_p$ and compute $Y = g^x$. The secret key is x and the public key is $(p, \mathbb{G}, \mathbb{G}_T, e, g, g_0, h_1, \ldots, h_L, Y)$.

Sign. Given message $M_1, \ldots, M_L \in \mathbb{Z}_p$, randomly select $w, r \in \mathbb{Z}_p$ and compute $A = (\prod_{1 \leq j \leq L} h_j^{M_j} \cdot g_0^r \cdot g)^{1/(x+w)}$. The signature is (A, w, r).

Verify. Given the signatures (A, w, r) on message $M_1, \ldots, M_L$, check $e(A, Yg^w) = e(\Pi_{1 \le j \le L} h_j^{M_j} \cdot g_0^r \cdot g, g)$.

The BBS+ signature is proved to be unforgeable against adaptively chosen message attack under the q-SDH assumption.

2.4 F-Secure BB Signature

We also adopt F-secure BB signature proposed in [8] in our scheme. This scheme is proposed as following:

Setup. Select bilinear groups $\mathbb{G}, \mathbb{G}_T$ with prime order p and a bilinear map e. Select $h, \tilde{h} \in \mathbb{G}$.

KeyGen. Select $\tilde{x}, \hat{x} \in \mathbb{Z}_p$ and compute $\tilde{Y} = h^{\tilde{x}}, \hat{Y} = h^{\hat{x}}$. The secret key is $(\tilde{x}, \hat{x})$ and the public key is $(p, \mathbb{G}, \mathbb{G}_T, e, g, h, \tilde{Y}, \hat{Y})$.

Sign. Given message $m \in \mathbb{Z}_p$, randomly select $\mu \in \mathbb{Z}_p - \{\frac{\tilde{x}-m}{\hat{x}}\}$ and compute $S = h^{1/(\tilde{x}+m+\hat{x}\mu)}, T = \hat{Y}^{\mu}, U = \hat{h}^{\mu}$. The signature is (S, T, U).

Verify. Given the signatures (S, T, U) on message M, check $e(S, \tilde{Y}h^m T) = e(h, h)$ and $e(\tilde{h}, T) = e(U, \hat{x})$.

Besides the normal unforgeablity, this signature system has a property called F-security defined as below: Define bijection F as $F(M) = (h^M, \tilde{h}^M)$ for Message M. The F-security of this signature means that no adversary can output a tuple $(F(M), \sigma)$ where σ is a valid signature on M unless he previously obtained a signature on message M. The F-security of FBB signature above can be proved under the q-HSDH and q-TDH assumptions.

2.5 Proofs of Knowledge

To prove the string attributes and achieve non-transferability we adopt zero-knowledge proofs of knowledge(POKs) on representations. By using this, the prover can prove the knowledge of a representation that for some $C, g_1, g_2, \ldots, g_n \in G$, he knows $x_1, \ldots, x_n$ satisfy the equation $C = g_1^{x_1} \cdots g_n^{x_n}$, to simplify the description, we denote this proof as $POK\{(x_1, \ldots, x_n) | C = g_1^{x_1} \cdots g_n^{x_n}\}$. Moreover, the POKs can be extended to prove multiple exponents equal. For prime-order groups which we used in this paper, there exists a knowledge extractor which can extract these quantities from a successful prover.

3 Scheme Construction

3.1 Anonymous Credential with Efficient Attribute Attestation

By using the signature schemes mentioned above and attribute-based cryptography, we propose an anonymous credential scheme with efficient attribute attestation, the concrete scheme are defined as follow:

Setup. Let $\mathbb{G}$ and $\mathbb{G}_T$ be cyclic groups of the prime order p, where g is a generator of $\mathbb{G}$. $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ be a bilinear mapping from $\mathbb{G}$ to $\mathbb{G}_T$. First, choose n as the maximum number of attributes in single threshold supported by the system, then randomly select a number $\omega_i \in \mathbb{Z}_p^*$ for each probably used attribute, furthermore, $n-1$ additional dummy attributes d_i are chosen from $\mathbb{Z}_p^*$ as well, these dummy attributes would never be issued.

Secondly, randomly select $g_0, g_F, h_0, h_1, \ldots, h_L, h, \widehat{g}, \widehat{h} \in \mathbb{G}$, L is the number of string attributes. Finally, the public parameters are defined as params=$\{ \mathbb{G}, \mathbb{G}_T, e, g_0, g_F, h_0, h_1, \ldots, h_L, h, \widehat{g}, \widehat{h}, \Omega = \{\omega_i\}, \mathbb{D} = \{d_j\} \}$.

IssuerGen. To generate the key pair of issuer ,firstly randomly choose $x, x_0, \widetilde{x}, \widehat{x} \in \mathbb{Z}_p^*$ and compute $Y_0 = g^{x_0}, \widetilde{Y} = g^{\widetilde{x}}, \widehat{Y} = g^{\widehat{x}}, g^x, g^{x^2}, \ldots, g^{x^{2n-1}}, h^x, h^{x^2}, \ldots, h^{x^{n-1}}$. The issuer secret key is $isk = \{x, x_0, \widetilde{x}, \widehat{x}\}$ and public key $ipk = \{Y_0 = g^{x_0}, \widetilde{Y} = g^{\widetilde{x}}, \widehat{Y} = g^{\widehat{x}}, (g^x, g^{x^2}, \ldots, g^{x^{2n-1}}), (h^x, h^{x^2}, \ldots, h^{x^{n-1}})\}$.

CreIssue. Suppose the user has some secret infomation f, the corresponding public key is $F = g_F^f$, the finite attributes contain in the credential is Ω_u, as well as L string attributes $M_1, \ldots, M_L$. The issuer proceeds as follow:

1. Check the validity of F and all attributes.

2. Randomly choose a token $g_u = g^u \in \mathbb{G}$ and compute $U_i = g_u^{1/(x+\omega_{U_i})}$ for each finite attribute ω_{U_i}.

3. Use F-secure BB signature with secret key $\widetilde{x}, \widehat{x}$ to generate a signature on message u, the signature is defined as $\sigma_{FBB} = (S, T, U) = (g^{1/\widetilde{x}+u+\widehat{x}\mu}, \widehat{Y}^{\mu}, \widehat{h}^{\mu})$, additionally, issuer computes $h_u = \widehat{h}^u$.

4. Use the BBS+ Signature scheme with secret key x_0 to sign string attributes $M_1, \ldots, M_L$ together with f and u, the signature is $\sigma_{BBS} = (A, w, r) = ((F g^u \prod_{j=1}^{L} h_j^{M_j} g_0^r g)^{1/(x_0+w)}, w, r)$.

5. Output the credential $cre = g_u, U_i, S, T, U, h_u, A, w, r$.

AttributeProve. When user wants to prove that he has a valid credential which contains string attributes $\{SA\} = S_1, \ldots, S_j$ and his finite attributes satisfied with the predicate $\Upsilon = (t, \mathbb{A})$, which is a (t, k)threshold for an attributes set $\mathbb{A}(1 \le t \le k = |\mathbb{A}| \le n)$ he proceeds as follow:

1. Firstly choose a subset $\Omega_u^{'}$ he owns that $\Upsilon(\Omega_u^{'}) = 1$, where $\Omega_u^{'} \subseteq \mathbb{A} \cap \Omega_u$ and $|\Omega_u^{'}| = t$. Then select the first $n+t-k-1$ attributes from $\mathbb{D}$, for $t \le k$, the size of this set is less than $n-1$, denote it as $\mathbb{D}_{n+t-k-1}$.

2. By using the aggregate algorithm in [6],it is possible to compute

$$A_1 = g_u^{\frac{1}{\prod_{\omega_{U_i} \in \Omega_u^{'}} (x+\omega_{U_i})}}$$

Then, for $|\mathbb{D}_{n+t-k-1} \cup (\mathbb{A} \setminus \Omega_u^{'})| = (n+t-k-1) + (k-t) = n-1$, user could use $(g, g^x, g^{x^2}, \ldots, g^{x^{n-1}}), (h, h^x, h^{x^2}, \ldots, h^{x^{n-1}})$ to compute

$$A_2 = g^{\prod_{\omega \in \mathbb{D}_{n+t-k-1} \cup (\mathbb{A} \setminus \Omega_u^{'})} (x+\omega)}, A_3 = h^{\prod_{\omega \in \mathbb{D}_{n+t-k-1} \cap (A \setminus \Omega_u^{'})} (x+\omega)}$$

. 3. Then randomly choose $r, s \in \mathbb{Z}_p^*$ and output the proof as $\Pi = (\pi_1, \pi_2, \pi_3, \pi_4) = (A_1^{rs}, A_2^r, A_3^r, g_u^s)$.

4. Randomly select $\rho_\pi, \rho_A, \rho_S, \rho_T, \rho_U, \rho_H \in \mathbb{Z}_p^*$ and compute commitments $C_\pi = g_u\widehat{g}^{\rho_\pi}, C_A = A\widehat{g}^{\rho_A}, C_S = S\widehat{g}^{\rho_S}, C_T = T\widehat{g}^{\rho_T}, C_U = U\widehat{g}^{\rho_U}, C_H = h_u\widehat{g}^{\rho_H}$.
5. Then randomly select $\rho_w, \rho^{'}\mathbb{Z}_p^*$, sets $\alpha = \rho_A w, \zeta = \rho_S\rho_\pi$ and $\xi = \rho_S\rho_T$. Compute auxiliary commitments $C_w = g^w\widehat{g}^{\rho_w}, C_{\rho_S} = g^{\rho_S}\widehat{g}^{\rho^{'}}$ and set $\rho_\alpha = \rho_w\rho_A, \rho_\zeta = \rho^{'}\rho_\pi, \rho_\xi = \rho^{'}\rho_T$.
6. Finally, the user sends $\Pi, C_\pi, C_A, C_S, C_T, C_U, C_H, C_w, C_{\rho_S}$ to the verifier and use proofs of knowledge on representations to generate the following proofs and send it to the verifier:

$$POK(\rho_\pi, \rho_A, \rho_S, \rho_T, \rho_U, \rho_H, \rho_w, \rho^{'}, \alpha, \zeta, \xi, s, r, w, f, M_k) :$$

$$e(C_A, Y_0)e(\prod_{1\le k\le j, M_k\in\{SA\}} h_k^{M_k}g, g)^{-1} = \{\prod_{1\le k\le j, M_k\notin\{SA\}} e(h_k, g)^{M_k}\}$$

$$e(g_F, g)^f e(C_A, g)^{-w} e(\pi_4, g)^{1/s} e(g_0, g)^r e(\widehat{g}, Y_0)^{\rho_A} e(\widehat{g}, g)^\alpha \quad (1)$$

$$e(C_S, \widetilde{Y}C_\pi C_T)e(g, g)^{-1} = e(\widehat{g}, \widetilde{Y}C_\pi C_T)^{\rho_s} e(C_S, \widehat{g})^{\rho_\pi+\rho_T} e(\widehat{g}, \widehat{g})^{-\zeta-\xi} \quad (2)$$

$$e(\widehat{h}, C_T)e(C_U, \widehat{Y})^{-1} = e(\widehat{h}, \widehat{g})^{\rho_T} e(\widehat{g}, \widehat{Y})^{-\rho_U} \quad (3)$$

$$e(\widehat{h}, C_\pi)e(C_H, g)^{-1} = e(\widehat{h}, \widehat{g})^{\rho_\pi} e(\widehat{g}, g)^{-\rho_H} \quad (4)$$

$$C_\pi = \pi_4^{1/s}\widehat{g}^{\rho_\pi} \quad (5)$$

$$C_w = g^w\widehat{g}^{\rho_w}, 1 = C_w^{\rho_A} g^{-\alpha}\widehat{g}^{-\rho_\alpha} \quad (6)$$

$$C_{\rho_S} = g^{\rho_S}\widehat{g}^{\rho^{'}}, 1 = C_{\rho_S}^{\rho_\pi} g^{-\zeta}\widehat{g}^{-\rho_\zeta}, 1 = C_{\rho_S}^{\rho_T} g^{-\xi}\widehat{g}^{-\rho_\xi} \quad (7)$$

Verify. After receiving the attribute attestation from the user, verifier verifies the correctness of the proofs of knowledge above at first. Then the verifier checks the following equation:

$$e(\pi_4, \pi_2) = e(\pi_1, g^{\prod_{\omega\in\mathbb{D}_{n+t-k-1}\cup\mathbb{A}}(x+\omega)})$$

and

$$e(\pi_2, h) = e(\pi_3, g)$$

if all of the above are correct, accept the attestation, otherwise, reject it.

3.2 Security Results

Privacy. In the Attribute Prove procedure, the verifier receives following messages: $(\Pi, C_\pi, C_A, C_S, C_T, C_U, C_H, C_w, C_{\rho_S})$, for the commitments C_π, C_A, C_S, $C_T, C_U, C_H, C_w, C_{\rho_S}$, which is randomized by $\rho_\pi, \rho_A, \rho_S, \rho_T, \rho_U, \rho_H, \rho_w, \rho^{'}$ and the zero knowledge property of $POKs$, these values contain no extra information about the user and is unlinkable. Then we consider the values in Π, for randomly chosen r and s, the value of π_1 and π_4 are uniformly distributed in group $\mathbb{G}$. Furthermore, when π_1 and π_4 are determined, the value of π_2 and π_3 are uniquely determined by threshold parameter n, k and attributes in set $\mathbb{A}$, which is only dependent with the predicate Υ, contains no information about user attributes and identity, too. From the above analysis, we can see that our scheme has full privacy and unlinkablity for user identity and attributes.

Unforgeabilty. For the unforgeability, we have the following theorem:

Theorem 1. The Attribute attestation protocol is a proof of knowledge of a modified BBS+ signature (A, w, r) on secret f, string type attributes $M_1, \cdots, M_L$, and the finite type of attributes is unforgeable under q-HPDH assumption.

Proof. The proof of Theorem 1 is described in our full paper.

4 Efficiency Results

In this section, we will compare the efficiency of our system with the pairing based system using accumulators in [5]. We use the same environment in [5] which described a common eID system, here is the parameter setting:

L: the total number of string attribute types. $\tilde{L}$: the total number of finite attribute types. n: the total number of finite attribute values. k: the number of attributes referenced in a proof. In addition, our system uses the following parameters: N: the upper limit of threshold parameter the system supports. According to paper [5], in an eID system, an approximate value of those parameters are $L = 5, \tilde{L} = 40, n = 1000$ to 10000 and $k = 10$, for there is no N in that scheme, we set it to 20, which is sufficient for normal attribute-based access control.

4.1 Computational and Communication Complexity

According to paper [5], we consider the computational complexity based on the number of exponentiations and pairings. Both our scheme and the scheme in [5] can achieve computational complexity independent with the total number of finite attribute types $\tilde{L}$, which is an advantage to scheme [4]. Although our scheme takes some more exponentiations than [5] for the proof of AND relation, but this is because our scheme is designed for the general threshold predicate with complexity independent to the type of predicate.

Then we compare the communication complexity of the attributes proof. The proof length in both our scheme and [5] are independent with the number of finite attributes in the predicate. For AND relation, our scheme has 3 more group components than scheme [5], which is roughly equal, but in the situation of any other relations, our scheme is more efficient. The concrete results can be found in our full paper.

4.2 Storage Data Size

For scheme [5] there was a main problem that for each probably used attribute in the system, the user has to store a tuple of corresponding public parameters, from the discussion of [5], this part of data consists with about 6000 to 60000 elements and would take a space of 200KB to 2MB. For common used eID

cards, this size is too large. When $L = 5, \tilde{L} = 40, n = 1000, k = 10, N = 20$, the total user data size of our scheme, which contains public parameters and user credentials, contains about 120 group elements. Compared with 6000 to 60000 elements in [5], our scheme can save more than 97 percents storage cost in user device.

5 Conclusion

In this paper, we considered a new way to build anonymous credentials with efficient anonymous attributes proofs using Attribute-based signature and proposed a concrete anonymous credential scheme. By using this new construction idea, our scheme could realize constant complexity attribute proofs while support more flexible threshold relations. Furthermore, our scheme solves the problem of the oversized public key in [5]. Finally, our scheme could be extended to support general attributes predicates, which are hard and inconvenient to realize in common anonymous credentials.

References

1. Maji, H., Prabhakaran, M., Rosulek, M.: Attribute based signatures: Achieving attribute privacy and collusion-resistance (2008), `http://eprint.iacr.org/2008/328`
2. Maji, H., Prabhakaran, M., Rosulek, M.: Attribute-Based Signatures. In: Kiayias, A. (ed.) CT-RSA 2011. LNCS, vol. 6558, pp. 376–392. Springer, Heidelberg (2011)
3. Camenisch, J., Lysyanskaya, A.: Dynamic Accumulators and Application to Efficient Revocation of Anonymous Credentials. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 61–76. Springer, Heidelberg (2002)
4. Camenisch, J., Gross, T.: Efficient attributes for anonymous credentials. In: ACM-CCS 2008, pp. 345–356 (2008)
5. Sudarsono, A., Nakanishi, T., Funabiki, N.: Efficient Proofs of Attributes in Pairing-Based Anonymous Credential System. In: Fischer-Hübner, S., Hopper, N. (eds.) PETS 2011. LNCS, vol. 6794, pp. 246–263. Springer, Heidelberg (2011)
6. Delerablbee, C., Pointcheval, D.: Dynamic Threshold Public-Key Encryption. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 317–334. Springer, Heidelberg (2008)
7. Boneh, D., Boyen, X., Shacham, H.: Short Group Signatures. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 41–55. Springer, Heidelberg (2004)
8. Belenkiy, M., Chase, M., Kohlweiss, M., Lysyanskaya, A.: P-signatures and Noninteractive Anonymous Credentials. In: Canetti, R. (ed.) TCC 2008. LNCS, vol. 4948, pp. 356–374. Springer, Heidelberg (2008)
9. Boyen, X., Waters, B.: Full-Domain Subgroup Hiding and Constant-Size Group Signatures. In: Okamoto, T., Wang, X. (eds.) PKC 2007. LNCS, vol. 4450, pp. 1–15. Springer, Heidelberg (2007)