A New Approach to Group Signature Schemes

Xiangguo Cheng
School of Information Engineering, Qingdao University, Qingdao 266071, China
Email: chengxg@qdu.edu.cn

Chen Yang and Jia Yu
China Electronics Standardization Institute, Beijing 100007,China.
School of Information Engineering, Qingdao University, Qingdao 266071, China
Email: yangchenyf@163.com yujia@qdu.edu.cn

***Abstract*—This paper presents a new approach to group signature schemes. The advantage of this approach is that it provides a concurrent join and fast revocation to group members. Using this method, Based on the famous BLS signature scheme from bilinear pairings, we put forward a new group signature scheme. Due to the simple constructive method and the sound properties of bilinear pairings, it is shown that our scheme is very efficient. The proposed scheme is dynamic and has the advantages of short signature length. We analyze our scheme using the formal security notion of a dynamic group signature scheme and show that it satisfies the security properties of correctness, anonymity, traceability and Non-frameability.**

***Index Terms*—Group signature, Short signature, Revocation, Threshold signature, Multi-signature, Bilinear pairing**

I. Introduction

A group signature scheme, primitively proposed by Chaum and van Heyst [1], allows a group member of a given group to anonymously sign messages on behalf of the group and can be only opened by the group manager. Participants in a group signature scheme are a set of group members and a group manager. The roles of the group manager is to register new users by issuing membership certificates and revoke the anonymity by opening the signature in case of a later dispute. In some schemes the functions of the group manager can be split between two managers: an issuer and an Opener. This is a desirable property that allows distribution of trust.

Group signatures have many practical applications in which user anonymity is required such as anonymous credential systems [2], identity escrow [3], voting and bidding [4], and electronic cash systems. In early group signature schemes [1,5,6], the size of the group public key and the signature grew with the size of the group and so these schemes are impractical for large groups. Schemes with fixed size of group public key and signature length have been presented in [2,4,7-13].

We note that many group signature schemes are constructed by making use of two different ordinary signatures. One is used to generate membership certificates as part of the Join protocol and the other one, which is in fact a non-interactive version of zero-knowledge protocol, is used in the group signature generation algorithm. Consequently, the join of group members and the generation and verification of group signatures are very complicated. There are some efficient group signature schemes, such as [4], [9] and [10]. However, it is still an open problem to design a group signature scheme that is secure and as efficient as the regular signature scheme such as RSA or DSA.

This paper presents a new and practical approach to group signatures. The main idea of this approach is to split the Issuer's private key into two parts. One part is given to the user, and the other one is given to the Opener. If and only if with the help of the Opener, a group member is able to generate a signature. The Corresponding public key of the Issuer is regarded as the group public key. It is obvious that such a signature is anonymous. To trace a group signature, the Opener need only store the identity of the signer when providing him signing help. Using the above constructive method, based on the famous BLS signature scheme from bilinear pairings given by Boneh et al. [14], we put forward a new group signature scheme. Due to the simple constructive method and the sound properties of bilinear pairings, it is shown that our scheme is very simple, efficient and satisfies the security properties of correctness, anonymity, traceability and non-frameability under the formal security notion of a dynamic group signature scheme given by Bellare et al. [13].

The rest of this paper is organized as follows. Section 2 presents the model of a group signature scheme. Section 3 describes the new approach to group signature schemes. The new group signature scheme and its security analysis are given in Section 4. The advantages of our scheme are shown in Section 5. The last section is the conclusion of our paper.

II. Preliminaries

Let G_1 be a cyclic additive group generated by P, whose order is a prime p, and G_2 a cyclic multiplicative group of the same order.

Manuscript received May 4, 2010; revised August 7, 2010; accepted September 7, 2010.
Corresponding author: Xiangguo Cheng

doi:10.4304/jcp.6.4.812-817

Definition 1. A bilinear pairing is a computable map $e: G_1 \times G_1 \rightarrow G_2$ with the following two properties.

(1) Bilinearity: $e(mP_1, nP_2) = e(P_1, P_2)^{mn}$ for any $m, n \in Z_p$ and $P_1, P_2 \in G_1$.

(2) Non-degeneracy: There exists $P_1, P_2 \in G_1$ such that $e(P_1, P_2) \neq 1$. Which means that $e(P, P) \neq 1$ since P is a generator of G_1.

Assume that the Discrete Logarithm problem in both G_1 and G_2 is hard. Consider the following two problems in G_1.

(1) Computational Diffie-Hellman (CDH) problem: Given $P, mP, nP \in G_1$ for any unknown $m, n \in Z_p$, compute $mnP \in G_1$.

(2) Decisional Diffie-Hellman (DDH) problem: Given $P, mP, nP, lP \in G_1$ for any unknown $m, n, l \in Z_p$, decide whether $l \equiv mn \mod p$.

Both the CDH and DDH problems are generally considered to be hard in G_1. However, the DDH problem becomes easy with the help of bilinear pairing since $l \equiv mn \mod p$ if and only if $e(mP, nP) = e(P, lP)$.

Definition 2. We call G a Gap Diffie-Hellman (GDH) group if DDH problem is easy while CDH problem is hard in G.

The above discussion shows that bilinear pairing can help us to obtain GDH groups. Such groups can be found on super-singular elliptic curves or hyper-elliptic curves over the finite fields, and the bilinear pairings can be derived from the Weil or Tate pairings [14, 15].

Throughout this paper, the system parameters in our scheme are denoted as $SP = \{G_1, G_2, e, p, P, H\}$ and can be obtained by running a GDH parameters generator [14,15], where G_1, G_2, e, p and P are defined as above and $H: \{0,1\}^* \rightarrow Z_p^*$ is a cryptographic hash function.

We use the formal model of a dynamic group signature scheme that includes a dynamically changing membership and a separation of a group manager into an Issuer and an Opener presented in [13]. This section recalls this model. We first describe participants and procedures in this model, then describe oracles accessible to the adversaries and finally define the formal security requirements.

III. New Approach to Group Signatures

The manager of a given group in our approach is separated into two parts, the Issuer and the Opener. The main idea of this approach is that the public key of the Issuer is regarded as the public verification key of the group, and the corresponding private key is split into two parts by the Issuer. One part is given to the user who wants to join the group, and the other part, along with the user's identity, is given to the Opener. As a result, the user becomes a member of the group. A group signature is generated by a collaboration of a group member and the Opener (it is in fact a (2, 2) threshold signature plus some other signatures generated by a group member and the Opener). Neither the group member nor the Opener itself is able to generate a valid group signature. To sign a message, a group member has to ask for the Opener's help. To trace a group signature, the Opener need only store the identity of the signer when providing him signing help. The signature is anonymous since it is generated by a collaboration of the signer and the Opener. This constructive method makes a group signature scheme very easy to realize the join of group members, the open of group signatures, and the immediate revocation of group membership (The group membership of a user can be immediately revoked at any time if the Opener does not provide him signing help). Furthermore, it provides a simple method for converting a general digital signature scheme (such as RSA and DSA etc.) into a group signature scheme.

IV. New Group Signature Scheme

Using the abovementioned constructive method, based on the BLS signature scheme from bilinear pairings, a new and practical group signature scheme is proposed in this section. We describe our group signature scheme using the notation and the formal definition of group signature schemes given in [13]. The security analysis is also under this model. We refer the readers to [13] for the details.

A. Proposed Schemes

Our group signature scheme consists of two group managers (the Issuer and the Opener), and users with unique identities $i \in N$ (the set of positive integers). Each user can join the group and become a group member. The scheme is specified as a tuple $gs = (GKg, UKg, Join, Iss, Gsig, GVf, Open, Judge)$ of polynomial time algorithms which are defined as follows. We assume that the group size and the number of queries asked by the adversary are polynomially-bounded by the security parameter κ.

GKg: Given a security parameter κ, a trusted party (such as PKI) runs the GDH parameters generator to obtain the system parameters $SP = \{G_1, G_2, e, p, P, H\}$. Then it randomly chooses $x \in Z_p^*$ and computes $X = xP$. The group public key is denoted as $gpk = X$ and the issuing key is $ik = x$.

UKg: Given a user i, the trusted party randomly chooses $x_i \in Z_p^*$ and computes $X_i = x_i P$. The personal public key of user i is $upk[i] = X_i$ and his private key is $usk[i] = x_i$.

$Join, Iss$: The communication among the Issuer, the Opener and the set of users is assumed to be secure. To realize the join of user i, they collaborate to do as follows.

User i sends his identity $i \in N$ along with his public key X_i to the Issuer. The Issuer randomly chooses $x_i^{(u)} \in Z_p^*$, computes $x_i^{(o)} = (x - x_i^{(u)}) \mod p$ and $X_i^{(u)} = x_i^{(u)} P$. $x_i^{(u)}$ is sent to user i and $((i, X_i^{(u)}), x_i^{(o)})$ is sent to the Opener. The Opener makes an entry in the registration table reg : $reg[i]=(i,X_i^{(u)})$, where $(i,X_i^{(u)})$ is the group membership certificate of user i and $x_i^{(o)}$ is a secret key used by the Opener to help user i to realize the generation of group signatures. User i computes $y_i=(x_i^{(u)}+x_i)\bmod p$. After this protocol, user i becomes a group member and his group membership secret key is $gsk[i]=y_i$.

$GSig$: To generate a group signature on some message M , user i collaborates with the Opener to do the following work.

User i computes $\sigma_i^{(u)}=y_iH(M)$ and sends $(i,H(M),\sigma_i^{(u)})$ to the Opener.

The Opener first checks that the group membership of user i has not been revoked and checks whether $e(P,\sigma_i^{(u)})=e(Y_i,H(M))$, where $Y_i=X_i^{(u)}+X_i$. If so, it then randomly chooses $t_i\in Z_p^*$ and computes $\sigma_i^{(o)}=x_i^{(o)}H(M)$, $T_i=t_iP$, $\omega_i=t_iH(M)$, $\sigma_i=\sigma_i^{(u)}+\sigma_i^{(o)}+\omega_i$ and $S_i=X_i+T_i$. It stores $(i,\ H(M),\ t_i)$ and sets the signature on M to be $Sig_i=(\sigma_i,S_i)$.

GVf : The verifier accepts the group signature $Sig_i=(\sigma_i,S_i)$ on M if $e(P,\sigma_i)=e(X+S_i,H(M))$.

$Open$: To open a group signature, the Opener needs only consult the storage list to identify the original signer. It can provide the following proof to show that it is indeed generated by user i : Given a signature $Sig_i=(\sigma_i,S_i)$ on some message M , the Opener computes $\omega_i=t_iH(M)$, $\sigma_i^{(o)}=x_i^{(o)}H(M)$ and $\mu_i=\sigma_i-\sigma_i^{(o)}-\omega_i$.

$Judge$: μ_i is a BLS signature on M under the public key $Y_i=X_i^{(u)}+X_i$. Anyone can verify this signature by checking whether $e(P,\mu_i)=e(Y_i,H(M))$ hold. User i can not deny his signature since μ_i can be only produced by i whose group membership certificate and personal public key are $X_i^{(u)}$ and X_i , respectively.

Remarks: Our group signature is in fact a multi-signature of three individual signatures generated by the Opener and the group member. Given a signature $Sig_i=(\sigma_i,S_i)$ on some message M generated by group member i , where

$$\begin{aligned}\sigma_i&=\sigma_i^{(u)}+\sigma_i^{(o)}+\omega_i\\&=y_iH(M)+x_i^{(o)}H(M)+t_iH(M)\\&=(x_i^{(u)}+x_i)H(M)+x_i^{(o)}H(M)+t_iH(M)\\&=(x_i^{(u)}H(M)+x_i^{(o)}H(M))+x_iH(M)+t_iH(M)\\&=\pi_1+\pi_2+\pi_3\end{aligned}$$

and $S_i=X_i+T_i$. π_1 is a (2, 2) threshold BLS signature on M under the group public key X generated by the Opener and user i . π_2 and π_3 are two BLS signatures on M under the public key X_i and T_i generated by user i and the Opener, respectively. Neither the Opener nor user i is able to generate a valid signature π_1 since it has been shown in [16] that a threshold BLS signature is unforgeable. [16] also tell us that our group signature (as a multi-signature) is unforgeable.

B. Security Analysis

In the following we will show that our scheme satisfies the security requirements of correctness, anonymity, traceability and non-frameability.

Theorem 1. Our scheme satisfies the security property of correctness.

Proof. We first show the correctness of the signature. Given a signature $Sig_i=(\sigma_i,S_i)$ on some message M generated by an honest group member i , we have

$$\begin{aligned}e(P,\sigma_i)&=e(P,\sigma_i^{(u)}+\sigma_i^{(o)}+\omega_i)\\&=e(P,y_iH(M)+x_i^{(o)}H(M)+t_iH(M))\\&=e(P,(x_i^{(u)}+x_i)H(M)+x_i^{(o)}H(M)+t_iH(M))\\&=e(P,(x_i^{(u)}+x_i^{(o)})H(M)+x_iH(M)+t_iH(M))\\&=e(P,(x+x_i+t_i)H(M))\\&=e((x+x_i+t_i)P,H(M))\\&=e(X+X_i+T_i,\ H(M))\\&=e(X+S_i,\ H(M))\end{aligned}$$

Therefore, a valid group signature can be accepted by the Verify algorithm.

We then show the proof provided by the Opener is correct and can be used by any judger.

$$\begin{aligned}e(P,\mu_i)&=e(P,\sigma_i-\sigma_i^{(o)}-\omega_i)\\&=e(P,\sigma_i)e(P,\sigma_i^{(o)})^{-1}e(P,\omega_i)^{-1}\\&=e(X+S_i,H(M))e(X-X_i^{(u)},H(M))^{-1}e(T_i,H(M))^{-1}\\&=e(X+X_i+T_i-X+X_i^{(u)}-T_i,H(M))\\&=e(X_i^{(u)}+X_i,H(M))\\&=e(Y_i,H(M))\end{aligned}$$

Therefore, anyone can identify the correctness of the proof and group member i cannot deny his signing.

Theorem 2. The proposed scheme satisfies the security property of anonymity with the assumption that G_1 is a GDH group.

Proof. In the process of proving the anonymity of our group signature scheme, the adversary is provided with extremely strong attack capabilities, including the ability to fully corrupt the Issuer (The adversary is not only given the Issuer key ik , but also provided access to the $SndToI(\cdot,\cdot)$ oracle, which enables it to play the role of Issuer in interacting with users in the $Join$ protocol). The adversary is additionally allowed to obtain both the personal private key and the private signing key of any user via the $USK(\cdot)$ oracle; read the content of the registration table via the $RReg(\cdot)$; corrupt users and interact with the Issuer on their behalf via the $CrptU(\cdot,\cdot)$ and $SndToU(\cdot,\cdot)$ oracles; and obtain the identity of the signer of any signature except the challenge one via the $Open$ oracle.

In the following we will show that, if an adversary A can break the anonymity of our scheme, it is also able to solve an instance of CDH problem in G_1 .

A runs in two stages, a choose stage and a guess stage.

At the choose stage, A is given the personal private key and the group membership secret key of any group member and the Issuer's private key. It also has the power to add group members by running the Join protocol and revoke some group members by asking the Opener not to provide these members signing help. It is additionally given the access to *Open* oracle on group signatures of its choice. Proceeding adaptively, A generates some group signatures and manages to open these signatures via the *Open* oracle. Eventually algorithm A halts, outputting a message M and two honest group members i_0 and i_1.

At the guess stage, A is given a signature $Sig_{i_b} = (\sigma_{i_b}, S_{i_b})$ on M generated by group member i_b, where b is chosen randomly from $\{0,1\}$. The goal of A is to guess who is the original signer, i_0 or i_1. In this stage, A can still query the *Open* oracle, but not on the challenge signature.

If A wins in the guess stage, then the following discussion shows that it is also able to solve an instance of CDH problem.

Suppose that A can successfully guess that the signature is generated by group member i_b. Note that A knows the private key x_{i_b} of i_b and the Issuer's private key x. It can compute $\eta = xH(M)$, $\mu_{i_b} = x_{i_b}H(M)$, $\omega_{i_b} = \sigma_{i_b} - \eta - \mu_{i_b}$ and $T_{i_b} = S_{i_b} - X_{i_b}$. There exist $m, n \in Z_p^*$ such that $T_{i_b} = mP$ and $H(M) = nP$ since $T_{i_b}, H(M) \in G_1$. Note that

$$\begin{aligned} e(P,\omega_{i_b}) &= e(P, \sigma_{i_b} - \eta - \mu_{i_b}) \\ &= e(P,\sigma_{i_b}) e(P,\eta)^{-1} e(P,\mu_{i_b})^{-1} \\ &= e(X + S_{i_b}, H(M)) e(X, H(M))^{-1} e(X_{i_b}, H(M))^{-1} \\ &= e(X + S_{i_b} - X - X_{i_b}, H(M)) \\ &= e(T_{i_b}, H(M)) = e(mP, nP) = e(P, mnP) \end{aligned}$$

Due to the non-degeneracy of bilinear pairing, we have $\omega_{i_b} = mnP$. That is to say, A has generated a CDH tuple $(P, T_{i_b}, H(M), \omega_{i_b})$ in G_1. This is a contradiction to the fact that G_1 is a CDH group. Therefore, our group signature is indistinguishable and the proposed scheme has the security property of anonymity.

Theorem 3. The proposed scheme has the security property of traceability with the assumption that G_1 is a CDH group.

Proof. In the process of proving the traceability of our group signature scheme, the adversary A is allowed to create honest group members via $AddU(\cdot)$ oracle; obtain both the personal private key and the private signing key of any user via $USK(\cdot)$ oracle; read the content of the registration table via $RReg(\cdot)$ oracle and corrupt users and interact with the Issuer on their behalf via $CrptU(\cdot,\cdot)$ and $SndToI(\cdot,\cdot)$ oracles. A is additionally given the access to $GSign(\cdot,\cdot)$ and $Open(\cdot,\cdot)$ oracles.

In our scheme, group signatures are generated by the collaboration of group members and the Opener. The identity of the signer has been stored by the Opener at the time it provided him signing help. Therefore, the traceability in our scheme means that an adversary cannot generate a valid group signature without the help of the Opener.

Note that if an adversary A can generate a BLS signature forgery $\bar{\sigma}$ on some message $\bar{M}$ under the group public key X, it then randomly chooses $\bar{t} \in Z_p^*$ and computes $\bar{T} = \bar{t}P$, $\bar{\omega} = \bar{t}H(M)$. It is apparent that $\overline{Sig} = (\bar{\sigma} + \bar{\omega}, \bar{T})$ is a valid group signature on $\bar{M}$ that the Opener cannot open. Signatures under the group public key X are produced by collaborations of group members and the Opener in our scheme. They are in fact (2, 2) threshold signatures given in [16]. It is shown in [16] that, even if the group members are corrupted, the signatures are still unforgeable since the private share of the Opener is unknown to the adversary.

The above discussion tells us that our scheme has the security property of traceability if G_1 is a GDH group.

Theorem 4. The proposed scheme has the security property of non-frameability with the assumption that G_1 is a GDH group.

Proof. To prove the non-frameability of our scheme, we give an adversary A very strong attack capabilities, including the ability to fully corrupt the Issuer and the Opener, which means that A is not only given the Issuer's private key, but also allowed to access to the storage list of the Opener.

A is also given the capability of adding or revoking group members. The only unknown of A is the personal private keys of the honest group members. The non-frameability in our scheme means that an adversary cannot generate valid group signatures on behalf of some honest group members.

Given a signature $Sig_i = (\sigma_i, S_i)$ on message M generated by an honest group member i, where

$$\begin{aligned} \sigma_i &= \sigma_i^{(u)} + \sigma_i^{(o)} + \omega_i \\ &= (x_i^{(u)} + x_i)H(M) + x_i^{(o)}H(M) + t_iH(M) \\ &= (x_i^{(u)} + x_i^{(o)})H(M) + x_iH(M) + t_iH(M) \\ &= \pi_1 + \pi_2 + \pi_3 \end{aligned}$$

and $S_i = X_i + T_i$. The adversary A can easily generate π_1 since it knows the Issuer's private key x. π_3 can also be randomly generated by A. However, π_2 is a BLS signature on M under the public key X_i. It has been shown in [14] that such a signature is unforgeable if the CDH problem in G_1 is hard. Therefore, none except i can collaborate with the Opener to generate a valid group signature and the Opener can trace back to i. That is to say, our scheme has the security property of non-frameability.

V. Advantages

Compared with previous group signature schemes, our scheme has some advantages described as follows.

A. Concurrent Join and Fast Revocation

It is very easy in our scheme for a user to realize the "join" of the group and become a group member. Joining of users can be done concurrently at any time.

Note that the signature in our scheme is generated by a collaboration of a group member and the Opener. Neither the group member nor the Opener itself is able to generate a valid group signature. The group membership of a user can be immediately revoked at any time if the Opener does not provide him signing help.

It is also very easy for the Opener to open a signature. The Opener need only store the identity of the signer at the time it provides him signing help.

B. Signature Length and Computational Complexity

We compare the signature length and computational complexity of the proposed scheme to those of the previous schemes such as ACJT00 scheme [4], NS04 scheme [9] and BBS04 scheme [10].

TABLE I. COMPARISON OF SIZES (BYTES)

Scheme	gsk	gpk	Signature
ACJT00	370	768	1087
NS04	192	277	520
BBS04	43	86	192
Our Scheme	22	22	43

TABLE II. COMPARISON OF COMPLEXITY ($GSig$ / GVf)

Scheme	# SMul	# EAdd	# MExp	# pairing
NS04	11/8	5/5	8/10	0/3
BBS04	9/8	3/4	3/4	0/2
Our Scheme	4/0	3/1	0/0	0/1

We assume that our scheme is implemented using an elliptic curve or hyper-elliptic curve over a finite field Z_p, where p is a 170-bit prime, G_1 is a subgroup of an elliptic curve group or a Jacobian of a hyper-elliptic curve over a finite field of order p, elements in G_1 are 171-bit strings. G_2 is a subgroup of a finite field of size approximately 2^{1020} . A possible choice for these parameters can be found in [14, 15], where G_1 is derived from the curve $E/GF(3^l)$ defined by $y^2 = x^3 - x + 1$. We assume that system parameters in ACJT00 scheme are $\varepsilon = 1.1$, $l_p = 512$, $\kappa = 160$, $\lambda_1 = 838$, $\lambda_2 = 600$, $\gamma_1 = 1102$ and $\gamma_2 = 840$.

A group signature of NS04 scheme is composed of seven Z_p, five G_1 and two G_2 elements and that of BBS04 scheme comprises ten Z_p and six G_1 elements. In contrast, the signature in the proposed scheme is composed of six Z_p and one G_1 elements, and thus its signature length is the shortest among the other previous schemes. The result is summarized in TABLE I.

We also estimate the computational cost of our scheme and that of the previous schemes by the number of scalar multiplications and element additions in G_1, modular exponentiations in G_2 and the number of pairing operations required for $GSig$ and GVf, since these are the most costly computations. Although we cannot present a precise estimation of the computational cost of each operation since it depends on the choice of the groups G_1 and G_2. These computations can be done quite efficiently if we choose Tate pairing for e and adopt the computation tools described in [17]. We summarize the result in TABLE II. where "# SMul" , "# EAdd" , "# MExp" and "# pairing" are abbreviations of "the number of scalar multiplications", "the number of element additions", "the number of modular exponentiations" and "the number of pairing operations".

VI. CONCLUSIONS

In this paper, we have proposed a new approach to group signature schemes. Using this method, based on the famous BLS signature scheme, we presented a simple and practical group signature scheme and analyzed its security under the strong formal security notion of a dynamic group signature scheme. The advantage of this method is that it provides an easy way to realize the "join" of group members, the "open" of group signatures, and the immediate "revocation" of group memberships.

ACKNOWLEDGMENT

This work was supported in part by a grant from the National Natural Science Foundation of China (No.60703089), the Natural Science Foundation of Shandong Province of China (No.Y2008A03) and the Plan Foundation of Science and Technology of Shandong Provincial Education Department of China (No. J07YH06 and No. J08LJ02).

REFERENCES

[1] D. Chaum and E. van Heyst, "Group Signatures," *Advances in Eurocrypt'1991*, LNCS, vol. 547, Springer-Verlag, 1991,pp. 257-265.

[2] G. Ateniese and B. de Medeiros, "Efficient group signatures without trapdoors," *Advances in Asiacrypt'2003*, LNCS, Vol. 2894, Springer-Verlag, 2003, pp. 246-268.

[3] S. Kim, S. Park and D. Won, "Convertible group signatures," *Advances in Asiacrypt'1996*, LNCS, vol. 1163, Springer-Verlag, 1996, pp. 311-321.

[4] G. Ateniese, J. Camenisch , M. Joye and G. Tsudik, "A practical and provably secure coalition-resistant group signature scheme," *Advances in Crypto'2000*, LNCS, vol. 1880, Springer-Verlag, 2000, pp. 255-270.

[5] J. Camenisch and M. Stadler, "Efficient and generalized group signatures," *Advances in Eurocrypt'1997*, LNCS, vol. 1233, Springer-Verlag, 1997, pp. 465-479.

[6] L. Chen and T. P. Pedersen, "New group signature schemes," *Advances in Eurocrypt'1994*, LNCS, vol. 950, Springer-Verlag, 1994, pp. 171-181.

[7] J. Camenisch and M. Stadler, "Efficient group signature schemes for large groups," *Advances in Crypto'1997*, LNCS, vol. 1296, Springer-Verlag, 1997, pp. 410-424.

[8] J. Camenisch and M. Michels, "A group signature scheme with improved efficiency," *Advances in Asiacrypt'1998*, LNCS, vol. 1514, Springer-Verlag, 1998, pp. 160-174.

[9] L. Nguyen and R. Safavi-Naini, "Efficient and provably secure trapdoor-free group signature schemes from bilinear

pairings," *Advances in Asiacrypt'2004*, LNCS, vol. 3329, Springer-Verlag, 2004, pp. 372-386.

[10] D. Boneh, X. Boyen and H. Shacham, "Short group signatures," *Advances in Crypto'2004*, LNCS, vol. 3152, Springer-Verlag, 2004, pp. 41-55.

[11] J. Camenisch and M. Michels, "Separability and efficiency for generic group signature schemes," *Advances in Crypto'1999*, LNCS, vol. 1666, Springer-Verlag, 1999, pp. 413-430.

[12] M. Bellare, D. Micciancio and B. Warinschi, "Foundations of group signatures: formal definitions, simplified requirements, and a construction based on general assumptions," *Advances in Eurocrypt'2003*, LNCS, vol. 2656, Springer-Verlag, 2003, pp. 614-629.

[13] M. Bellare, H. Shi and C. Zhang, "Foundations of group signatures: the case of dynamic groups," *Topics in CT-RSA 2005*, LNCS, vol.3376, Springer-Verlag, 2005, pp.136-153.

[14] D. Boneh, B. Lynn and H. Shacham, "Short signatures from the Weil pairing," *Advances in Asiacrypt'2001*, LNCS, vol. 2248, Springer-Verlag, 2001, pp. 514-532.

[15] D. Boneh and M. Franklin, "Identity based encryption from the Weil pairing," *Advances in Crypto'2001*, LNCS, vol. 2139, Springer-Verlag, 2001, pp. 213-229.

[16] A. Boldyreva, "Efficient threshold signature, multi-signature and blind signature schemes based on the gap-Diffie-Hellman-group signature scheme," *Advances in PKC 2003*, LNCS, vol. 2567, Springer-Verlag, 2003, pp. 31-46.

[17] A. Menezes, C. van Oorschot and S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997, pp. 617-627.