Generalized ID-Based ElGamal Signatures with Message Recovery

Said Kalkan, Kamer Kaya, and Ali Aydın Selçuk

***Abstract*—Signature schemes with message recovery provide the feature that the message is recoverable from the signature and hence does not need to be transmitted separately. Recently a number of ID-based signature schemes with message recovery have been proposed. In this paper, we introduce the generalized ID-based ElGamal signatures with message recovery. The previously proposed ID-based signature schemes with message recovery turn out to be special instances of this generalized scheme. We also obtain several new ID-based signatures with message recovery from this generalized scheme which have not been explored before.**

***Index Terms*—bilinear pairing, ID-based signature, message recovery.**

I. Introduction

IN 1984, Shamir [10] introduced the concept of ID-based cryptography to simplify key management procedures in public key infrastructures. Following Joux's [6] discovery on how to utilize bilinear pairings in public key cryptosystems, Boneh and Franklin [3] proposed first practical ID-based encryption scheme in Crypto 2001. Since then, ID-based cryptography has been one of the most active research areas in cryptography and numerous ID-based encryption and signature schemes have been proposed that use bilinear pairings.

ID-based cryptography helps us to simplify the key management process in traditional public key infrastructures. In ID-based cryptography any public information such as e-mail address, name, etc., can be used as a public key. Since public keys are derived from publicly known information, their authenticity is established inherently and there is no need for certificates in ID-based cryptography. The private key for a given public key is generated by a trusted authority and is sent to the user over a secure channel.

In signature schemes with message recovery, the message is not transmitted together with the signature, and is recovered according to the verification process. This kind of signatures are used if the message length is short and bandwidth is a main concern.

In 1993, Nyberg and Rueppel [8] proposed an ElGamal-based [4] signature scheme with message recovery which was followed by several other proposals [9], [12], [1], [7]. The first ID-based signature with message recovery was proposed by Zhang et al. [13] in 2005. Tso et al. [11] proposed a more efficient scheme more recently in 2007.

In this paper, we introduce the concept of generalized ID-based ElGamal signatures with message recovery and show that the previously proposed signature schemes are special instances of this generalized scheme. The generalized scheme also yields many new ID-based signatures with message recovery that have not been explored before.

The rest of the paper is organized as follows: Background concepts including bilinear pairings and ElGamal signatures with message recovery are discussed in Section II. We describe the basic ID-based ElGamal signature with message recovery in Section III. In Section IV, we describe the generalizations of the basic scheme. We modify some of these schemes and produce more efficient signatures in Section V. We show how to embed previously proposed signatures into our generalized scheme in Section VI. The paper is concluded in Section VII.

S. Kalkan, K. Kaya, and Ali Aydın Selçuk are with the Department of Computer Engineering, Bilkent University, Ankara, 06800, Turkey (e-mail: {skalkan,kamer,selcuk}@cs.bilkent.edu.tr).

II. Background

In this section, we present the tools which will be used in the rest of the paper. We briefly discuss bilinear pairings, the basic ElGamal signature scheme with message recovery and its generalizations.

A. Bilinear Pairings

Let G_1 be a cyclic additive group of order q generated by P. Let G_2 be a cyclic multiplicative group of the same order. An admissible bilinear pairing is defined as $e : G_1 \times G_1 \rightarrow G_2$ with the following properties:

1) *Bilinearity:* $e(aR, bS) = e(R, S)^{ab}$ where $R, S \in G_1$ and $a, b \in \mathbb{Z}_q$. This can also be stated as $\forall R, S, T \in G_1$ $e(R + S, T) = e(R, T)e(S, T)$ and $e(R, S + T) = e(R, S)e(R, T)$
2) *Non-degeneracy:* The map e does not send all pairs in $G_1 \times G_1$ to the identity of G_2. That is $e(P, P) \neq 1$.
3) *Computability:* There exists an efficient algorithm to compute $e(R, S)$ for any $R, S \in G_1$

B. ElGamal Signature Scheme with Message Recovery

Nyberg and Rueppel showed that the ElGamal signatures can be extended to provide message recovery. The extension is done as follows: Let p be a large prime, q a divisor of $p-1$, and g an element in $\mathbb{Z}_p^*$ of order q. The user chooses $\alpha \in \mathbb{Z}_q$ as his private key and $\beta = g^\alpha \bmod p$ as his public key. To sign a message $m \in \mathbb{Z}_p$, the user first generates a random number $k \in_R \mathbb{Z}_q^*$. Then he computes:

$$r = mg^{-k} \bmod p$$
$$s = k^{-1}(1 + r\alpha) \bmod q$$

The (r, s) pair is the signature of message m. The equation,

$$1 = r\alpha + ks \bmod q \quad (1)$$

is called the signature equation and the message m can be recovered by computing $m = g^{s^{-1}}\beta^{rs^{-1}} r \bmod p$. We call this scheme as the basic ElGamal message recovery scheme.

Note that, in the above scheme computation of the signature and message recovery involve inversion of the elements in $\mathbb{Z}_q$. Nyberg and Rueppel showed that it is also possible to get a signature without inversions. Signature computation and verification can be done without inverses by changing the signature equation as:

$$s = -\alpha r + k \bmod q.$$

The message m can now be recovered as $m = g^s\beta^r r \bmod p$ without any inversions.

C. Generalized ElGamal Signatures with Message Recovery

Horster et al. [5] showed that many variations of the basic ElGamal message recovery scheme are possible by modifying the signature equation (1). One can use the general equation

$$A = \alpha B + kC \bmod q$$

to obtain a signature, where (A, B, C) is a permutation of the parameters $(1, r, s)$. The parameter r can be computed as $r = g^{-k}m$ or $r = d(m, g^k)$ with a suitable function $d : \mathbb{Z}_p^2 \to \mathbb{Z}_p$ where $d^{-1}(r, g^k) = m$. The message m can be recovered from the signature (r, s) by computing

$$m = d^{-1}(r, g^{AC^{-1}\beta^{-BC^{-1}}} \bmod p).$$

The consistency of m should be verified by checking if m satisfies a certain redundancy scheme as explained in Section III-A.

Different signature schemes can be obtained by using different coefficients instead of just using the permutations of $(1, r, s)$. The coefficients (A, B, C) can also be chosen as a permutation of $(1, r, rs)$ or $(1, s, rs)$. Additionally, the signs of (A, B, C) can be changed by multiplying them by ± 1.

The generalization can be extended further by choosing A, B, C as general functions of r, s. In that case one of the functions should be chosen as 1 to get efficient variants. Additionally, suitable functions should be chosen to guarantee solvability of the parameter s. To guarantee security, the parameters r, s have to occur in at least one of the three coefficients. Also, the insecure rs variant should be avoided.

An insecure rs variant occurs if (A, B, C) is taken as a permutation of $(1, 1, rs)$: For some message m, an attacker chooses a random $c \in_R \mathbb{Z}_q^*$ and substitutes it for rs and computes g^{-k} from the verification equation. Then he computes first r from g^{-k} and then computes s as $s = cr^{-1}$. The (r, s) pair will be a valid signature for the message m.

III. Basic ID-based ElGamal Signatures with Message Recovery

An ID-based signature scheme consists of four algorithms: Setup, Extract, Sign, and Verify. In Setup, the PKG, chooses a secret as the global secret key and publishes the global public system parameters. In Extract, the PKG verifies a user's identity and computes his private key. In Sign, the user signs a message by using his private key. Finally in Verify, the verifier verifies the signature and recovers the message by using the public parameters and the signer's identity.

An ID-based message recovery signature scheme can be obtained from the original ElGamal signature scheme as follows:

- Setup: Let G_1 be cyclic additive group of order q generated by P. Let G_2 be a cyclic multiplicative group of the same order and $e : G_1 \times G_1 \to G_2$ be an admissible bilinear pairing. The PKG chooses $s \in_R \mathbb{Z}_q^*$ as the global secret key and computes $P_{pub} = sP$ as the global public key. The PKG publishes system parameters $\langle G_1,\ G_2,\ e,\ P,\ P_{pub},\ H_1\rangle$ where H_1 is a secure hash function.

- Extract: PKG verifies the user's identity ID and computes $Q_{ID} = H_1(ID)$ and $S_{ID} = sQ_{ID}$ as user's public and private keys respectively.

- Sign: To sign a message $m \in \mathbb{Z}_q$, a user with his private key S_{ID}, first chooses $k \in_R \mathbb{Z}_q$, then computes:

$$r = e(P, P)^k \oplus m$$
$$U = k(P - rS_{ID})$$

  The signature for the message m is (kP_{pub}, r, U)

- Verify: Given ID, and a signature (kP_{pub}, r, U), the message can be recovered as:

$$m = r \oplus (e(U, P)e(Q_{ID}, kP_{pub})^r)$$

Correctness of the given scheme can be shown easily by using the bilinearity properties of e. Notice that if (kP_{pub}, r, U) is a valid signature for m then we have:

$$\begin{aligned} e(U, P)e(Q_{ID}, kP_{pub})^r &= e(k(P - rS_{ID}), P)e(Q_{ID}, kP_{pub})^r \\ &= e(kP - krS_{ID}, P)e(krS_{ID}, P) \\ &= e(kP, P) \\ &= e(P, P)^k \end{aligned} \quad (2)$$

A. Consistency Checking for the Message

In order to prevent a random (r, s) pair being accepted as a valid signature, consistency of the message should be checked with a given redundancy scheme. Abe and Okamoto [1] proposed such a redundancy encoding for their message recovery signature which can also be used in our scheme: Let $|q|$ denote the length of q in bits. Let $[m']^{k_1}$ denote the most significant k_1 bits of m' and $[m']_{k_2}$ denote the least significant k_2 bits of m'. Instead of computing r as $r = e(P, P)^k \oplus m$, first compute

$$m' = F_1(m) \| (F_2(F_1(m)) \oplus m),$$

where $F_1 : \{0,1\}^{k_2} \to \{0,1\}^{k_1}$ and $F_2 : \{0,1\}^{k_1} \to \{0,1\}^{k_2}$ are secure hash functions; and compute

$$r = e(P,P)^k \oplus m'.$$

Then the message m with length $|k_2|$ can be recovered as

$$m = [m']_{k_2} \oplus F_2([m']^{k_1}).$$

Consistency of m can be verified by checking $[m']^{k_1} \stackrel{?}{=} F_1(m)$. The advantage of using Abe and Okamoto's redundancy encoding is that F_1 and F_2 can be seen as random oracles so m' will be a random value independent from m.

IV. The Generalized ID-based Message Recovery Signatures

We can generalize the above signature scheme by using the generalized signing equation

$$A = S_{ID}B + kC \tag{3}$$

where (A,B,C) is a permutation of the parameters $(1,r,U)$. Note that, the variants where U is a coefficient of S_{ID} do not produce useful signing equations. Also note that, P and rP are used instead of 1 and r in cases where they need to be members of the elliptic curve group.

We get four variants by permuting the elements of $(1,r,U)$. The signing equation for these variants are:

$$P = rS_{ID} + k^{-1}U \tag{4}$$
$$U = rS_{ID} + kP \tag{5}$$
$$U = S_{ID} + krP \tag{6}$$
$$rP = S_{ID} + k^{-1}U \tag{7}$$

In (5) and (6) the signature for m will be (r,U) and we can recover m without any extra information. However, in (4) and (7) we need the value of kP_{pub} for verification, and the signature will be (kP_{pub}, r, U).

More variants can be generated by using different permutations. Instead of choosing (A,B,C) as a permutation of $(1,r,U)$, we can also choose them as a permutation of $(1,r,rU)$. Also, signs of A, B and C can be changed by multiplying them by ± 1. Note that, unlike the generalized ElGamal message recovery signatures, we cannot choose (A,B,C) as a permutation of $(1,U,rU)$, since we cannot extract U from the signing equation.

The verification equations and other details for these signatures are summarized in Table I. Group I lists the variants that are obtained by permuting $(1,r,U)$ and Group II lists the variants obtained by permuting $(1,r,rU)$. Group III is the secure $(1,1,rU)$ variant which is discussed in Section IV-B.

A. Generalized Partial Message Recovery Signatures

In the above signature schemes, length of the message is fixed. If Abe and Okamoto's redundancy encoding is used, then $|m| = k_2$. Here we show how one of the previous schemes can be modified to allow arbitrary length messages by splitting the message m into two parts called m_1 and m_2. The first part m_1 is of arbitrary length and is given with the signature (r,U). The second part m_2 has a fixed length and is recovered from the signature.

As an example, consider MR I.2 of Table I. To sign a message $m = m_1 \| m_2$ with $m_2 \in \mathbb{Z}_q$, a user with his private key S_{ID}, first chooses $k \in_R \mathbb{Z}_q$, then computes:

$$r = e(P,P)^k \oplus m_2$$
$$U = kP - m_1 r S_{ID}$$

The signature for the message m is (m_1, r, U). Note that, a general function $f(m_1, r)$ can be used instead of the product $m_1 r$.

To verify a given signature (m_1, r, U), the message can be recovered as:

$$m_2 = r \oplus (e(U,P)e(Q_{ID}, P_{pub})^{m_1 r})$$
$$m = m_1 \| m_2$$

Correctness of this scheme can easily be shown by using the bilinearity properties of e. Consistency of m should be verified by checking if m satisfies a certain redundancy scheme.

B. Security of the Signatures

Similar to the meta-ElGamal signature schemes with message recovery [5], generalized ID-based signatures with message recovery are generally secure except the insecure rU variants. These variants occur if (A,B,C) is either $(rU,1,1)$ or $(1,1,rU)$. Signing equations for these variants are:

$$rU = -S_{ID} + kP \tag{8}$$
$$P = S_{ID} + rk^{-1}U \tag{9}$$

In (8) the message m should satisfy the verification equation

$$m = r \oplus (e(U,P)^r e(Q_{ID}, P_{pub}))$$

This signature is not secure and the rU attack for this signature works as follows: For arbitrary message m, the adversary chooses $T \in_R G_1$. The random T will be used instead of rU so the adversary substitutes $e(T,P)$ for $e(U,P)^r$ and computes $e(P,P)^k$ as

$$e(P,P)^k = e(T,P)e(Q_{ID}, P_{pub})$$

Then he computes r as $r = e(P,P)^k \oplus m$. After that, he computes $U = r^{-1}C$. The (r,U) pair will be a valid signature for the message m.

The verification equation for the signature obtained from (9) is

$$m = r \oplus (e(U,P)^r e(Q_{ID}, kP_{pub}))$$

This signature seems to be secure and the rU attack does not work because the verification equation contains kP_{pub}. Therefore, an attacker cannot extract r from the verification equation.

No.	r	U	Signature	Message Recovery
MR I.1	$r = e(P,P)^k \oplus m$	$U = kP - krS_{ID}$	(kP_{pub}, r, U)	$m = r \oplus (e(U,P)e(Q_{ID}, kP_{pub})^r)$
MR I.2	$r = e(P,P)^k \oplus m$	$U = kP - rS_{ID}$	(r, U)	$m = r \oplus (e(U,P)e(Q_{ID}, P_{pub})^r)$
MR I.3	$r = e(P,P)^k \oplus m$	$U = krP - S_{ID}$	(r, U)	$m = r \oplus (e(U,P)^{r^{-1}} e(Q_{ID}, P_{pub})^{r^{-1}})$
MR I.4	$r = e(P,P)^k \oplus m$	$U = krP - kS_{ID}$	(kP_{pub}, r, U)	$m = r \oplus (e(U,P)^{r^{-1}} e(Q_{ID}, kP_{pub})^{r^{-1}})$
MR II.1	$r = e(P,P)^k \oplus m$	$U = r^{-1}kP - kS_{ID}$	(kP_{pub}, r, U)	$m = r \oplus (e(U,P)^r e(Q_{ID}, kP_{pub})^r)$
MR II.2	$r = e(P,P)^k \oplus m$	$U = r^{-1}kP - S_{ID}$	(r, U)	$m = r \oplus (e(U,P)^r e(Q_{ID}, P_{pub})^r)$
MR II.3	$r = e(P,P)^k \oplus m$	$U = kP - r^{-1}S_{ID}$	(r, U)	$m = r \oplus (e(U,P)e(Q_{ID}, P_{pub})^{r^{-1}})$
MR II.4	$r = e(P,P)^k \oplus m$	$U = kP - r^{-1}kS_{ID}$	(kP_{pub}, r, U)	$m = r \oplus (e(U,P)e(Q_{ID}, kP_{pub})^{r^{-1}})$
MR III.1	$r = e(P,P)^k \oplus m$	$U = r^{-1}k(P - S_{ID})$	(kP_{pub}, r, U)	$m = r \oplus (e(U,P)^r e(Q_{ID}, kP_{pub}))$

TABLE I
THE GENERALIZED ID-BASED ELGAMAL SIGNATURES WITH MESSAGE RECOVERY.

V. MORE EFFICIENT SIGNATURES

Computing a signature requires one or two scalar multiplications in G_1 depending on how the signature equation is defined, as well as an exponentiation in G_2. The value $e(P,P)$ is fixed and can be precomputed, so pairing evaluation is not needed to sign a message.

The cost of verifying a signature will be dominated by the pairing computations, which is the most expensive operation. Two pairing computations are needed to verify a signature. Note that, in some of the proposed schemes (MR I.2, MR I.3, MR II.2, MR II.3), the value $e(Q_{ID}, P_{pub})$ is used, which is fixed for a particular user and needs to be computed only once for each user.

The number of pairing operations can be reduced to one by changing the definitions of S_{ID} and Q_{ID} as in [2]. If we define

$$Q_{ID} = (H_1(ID) + s)P$$
$$S_{ID} = (H_1(ID) + s)^{-1}P,$$

the number of pairing evaluations can be reduced to one. Note that, Q_{ID} can be computed by anyone, since the value of sP is public, but S_{ID} cannot be computed without knowing the value of s.

We can get efficient variants by changing the definitions of S_{ID} and Q_{ID} in four of the proposed schemes. These schemes are MR I.2, MR I.3, MR II.2, MR II.3 of Table I. The computation of r should also be changed to increase the efficiency. Instead of computing r as $r = e(P,P)^k \oplus m$, r will be computed as

$$r = e(P, Q_{ID})^k \oplus m$$

This modification does not affect the efficiency of signature computation, since the value $e(P, Q_{ID})$ can be precomputed by the sender.

As an example, consider the modified version of MR I.2 where $U = kP - rS_{ID}$. The message m can be recovered from the signature (r, U) as,

$$m = r \oplus (e(U, Q_{ID})e(P,P)^r).$$

The verification equations and other details of the efficient versions of MR I.2, MR I.3, MR II.2, MR II.3 modified in this fashion are given in Group IV of Table II.

Further variants with a reduced signing cost can be obtained by modifying the generalized signature equation as,

$$A = BS_{ID} + kCS_{ID}. \quad (10)$$

Note that, this kind of generalization is not possible over the basic ElGamal signatures, because when k and α are used together, we cannot extract s from the signing equation.

By the help of bilinear pairings we can extract U from the signing equation (10), if U is in A's position. We can get four more efficient variants whose signing equations are:

$$U = (k + r)S_{ID}$$
$$U = (1 + kr)S_{ID}$$
$$rU = (k + r)S_{ID}$$
$$rU = (1 + kr)S_{ID}$$

As an example, in the first scheme where $U = (k + r)S_{ID}$, the message m can be recovered from the signature (r, U) as,

$$m = r \oplus (e(U, Q_{ID})e(P,P)^{-r})).$$

The verification equations and other details of these signatures are given in Group V of Table II.

VI. EMBEDDING PREVIOUSLY KNOWN ID-BASED MESSAGE RECOVERY SIGNATURES

Recently two ID-based message recovery signature schemes have been proposed. These signatures [13], [11] can be seen as special instances of our generalized scheme.

In Zhang et al.'s scheme [13], the signature (r, U) for the message m is computed as

$$m' = F_1(m) \| (F_2(F_1(m)) \oplus m)$$
$$r = H_2(e(P,P)^k) + m' \bmod q$$
$$U = kP - rS_{ID}$$

where H_2 is a secure hash function. Zhang et al.'s scheme is equivalent to MR I.3 of Table I, where a hash function H_2 and Abe and Okamoto's redundancy encoding is used with a slightly different computation of r.

In Tso et al.'s scheme [11], the signature (r, U) for the message m is computed as

$$m' = F_1(m) \| (F_2(F_1(m)) \oplus m)$$
$$r = H_2(e(P,P)^k) \oplus m'$$
$$U = (k + r)S_{ID}$$

No.	r	U	Signature	Message Recovery
MR IV.1	$r = e(P, Q_{ID})^k \oplus m$	$U = kP - rS_{ID}$	(r, U)	$m = r \oplus (e(U, Q_{ID})e(P, P)^r)$
MR IV.2	$r = e(P, Q_{ID})^k \oplus m$	$U = krP - S_{ID}$	(r, U)	$m = r \oplus (e(U, Q_{ID})^{r^{-1}} e(P, P)^{r^{-1}})$
MR IV.3	$r = e(P, Q_{ID})^k \oplus m$	$U = r^{-1}kP - S_{ID}$	(r, U)	$m = r \oplus (e(U, Q_{ID})^r e(P, P)^r)$
MR IV.4	$r = e(P, Q_{ID})^k \oplus m$	$U = kP - r^{-1}S_{ID}$	(r, U)	$m = r \oplus (e(U, Q_{ID})e(P, P)^{r^{-1}})$
MR V.1	$r = e(P, P)^k \oplus m$	$U = (k + r)S_{ID}$	(r, U)	$m = r \oplus (e(U, Q_{ID})e(P, P)^{-r})$
MR V.2	$r = e(P, P)^k \oplus m$	$U = (1 + kr)S_{ID}$	(r, U)	$m = r \oplus (e(U, Q_{ID})^{r^{-1}} e(P, P)^{-r^{-1}})$
MR V.3	$r = e(P, P)^k \oplus m$	$U = r^{-1}(k + r)S_{ID}$	(r, U)	$m = r \oplus (e(U, Q_{ID})^r e(P, P)^{-r})$
MR V.4	$r = e(P, P)^k \oplus m$	$U = r^{-1}(1 + kr)S_{ID}$	(r, U)	$m = r \oplus (e(U, Q_{ID})e(P, P)^{-r^{-1}})$

TABLE II
EFFICIENT ID-BASED SIGNATURES WITH MESSAGE RECOVERY.

where H_2 is a secure hash function. Tso et al.'s scheme is equivalent to MR IV.1 of Table II where a hash function H_2 and Abe and Okamoto's redundancy encoding is used.

VII. CONCLUSION

In this paper, ID-based signatures with message recovery are investigated. We showed how the basic ElGamal signature with message recovery can be converted to an ID-based signature with message recovery. We extended our ID-based signature scheme into a generalized ID-based message recovery signature as in the work of Horster et al. [5] on basic ElGamal signatures with message recovery. We also presented some original variants which were not possible in the non-ID-based setting. Then, we modified some of our signatures to get more efficient signature schemes.

The generalized ID-based message recovery signature scheme we described provides a unified framework for ID-based ElGamal signatures with message recovery. The two ID-based message recovery signatures in the literature [13], [11] can be seen as special instances of the generalized scheme. This unified framework also yields many new ID-based signatures with message recovery that have not been explored before.

Among the proposed schemes, Group IV and Group V are the most efficient signatures, with just one pairing operation needed in signature verification. Group V has the further advantage of reducing the cost of the signature operation by one scalar multiplication in the elliptic curve group G_1.

REFERENCES

[1] M. Abe and T. Okamoto. A signature scheme with message recovery as secure as discrete logarithm. In *Proc. of ASIACRYPT'99*, volume 1716 of *LNCS*, pages 378–389. Springer-Verlag, 1999.

[2] P. S. L. M. Barreto, B. Libert, N.McCullagh, and J. Quisquater. Efficient and provably-secure identity-based signatures and signcryption from bilinear maps. In *Proc. of ASIACRYPT'05*, volume 3778 of *LNCS*, pages 515–532, 2005.

[3] D. Boneh and M. Franklin. Identity-based encryption from the Weil pairing. In *Proc. of CRYPTO'05*, volume 2139 of *LNCS*, pages 213–229. Springer-Verlag, 2001.

[4] T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Trans. Information Theory*, 31(4):469–472, 1985.

[5] P. Horster, M. Michels, and H. Petersen. Meta signature schemes giving message recovery based on the discrete logarithm problem. In *Proc. of 2nd Int. Workshop on IT-Security*, Vienna, 1994.

[6] A. Joux. A one round protocol for tripartite Diffie-Hellman. In *Proc. of ANTS-IV*, volume 1838 of *LNCS*, pages 385–394, 2000.

[7] A. Miyaji. A message recovery signature scheme equivalent to dsa giving message recovery. In *Proc. of ASIACRYPT'96*, volume 1163 of *LNCS*, pages 1–14, 1996.

[8] K. Nyberg and R. A. Rueppel. A new signature scheme based on the dsa giving message recovery. In *Proc. of 1st ACM conference on communication and computer security*, pages 58–61, 1993.

[9] K. Nyberg and R. A. Rueppel. Message recovery for signature schemes based on the discrete logarithm problem. In *Proc. of EUROCRYPT'94*, volume 950 of *LNCS*, pages 182–193, 1995.

[10] A. Shamir. Identity-based cryptosystems and signature schemes. In *Proc. of CRYPTO'84*, volume 196 of *LNCS*, pages 47–53. Springer-Verlag, 1984.

[11] R. Tso, C. Gu, T. Okamoto, and E. Okamoto. An efficient ID-based digital signature with message recovery based on pairing. http://citeseer.ist.psu.edu/tso06efficient.html.

[12] C. Y. Yeun. Digital signature with message recovery and authenticated encryption (signcryption)– a comparison. In *IMA - Cryptography and Coding'99*, volume 1746 of *LNCS*, pages 307–312, 1999.

[13] F. Zhang, W. Susilo, and Y. Mu. Identity-based partial message recovery signatures. In *Financial Cryptography'05*, volume 3570 of *LNCS*, pages 45–56, 2005.