Generalized ID-Based ElGamal Signatures

Said Kalkan
Department of Computer Engineering
Bilkent University
Ankara, 06800, Turkey
Email: skalkan@cs.bilkent.edu.tr

Kamer Kaya
Department of Computer Engineering
Bilkent University
Ankara, 06800, Turkey
Email: kamer@cs.bilkent.edu.tr

Ali Aydın Selçuk
Department of Computer Engineering
Bilkent University
Ankara, 06800, Turkey
Email: selcuk@cs.bilkent.edu.tr

***Abstract*— ID-based cryptography has been a very active area of research in cryptography since bilinear pairings were introduced as a cryptographic tool, and there have been many proposals for ID-based signatures recently. In this paper, we introduce the concept of generalized ID-based ElGamal signatures and show that most of the proposed ID-based signature schemes in the literature are special instances of this generalized scheme. We also obtain numerous new signatures from this generalized scheme which have not been proposed before.**

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 [7] discovery on how to utilize bilinear pairings in public key cryptosystems, Boneh and Franklin [2] 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.

Recently, there has been many proposals for ID-based signatures [9], [11], [1], [8], [5], [3] and most of these schemes, in one way or the other, have been based on the ElGamal signature algorithm [4]. In this paper, we show that most of these proposals are in fact special instances of a more general concept which we call the generalized ID-based ElGamal signature. Besides providing a unified framework for previously proposed ID-based signatures, the generalized scheme also yields many new ID-based signatures that have not been explored before.

The rest of the paper is organized as follows: Background concepts including bilinear pairings and ElGamal signatures are discussed in Section II. We explain how to convert the original ElGamal signature into an ID-based signature scheme in Section III. We use the ideas of Horster et al. [6] and generalize the basic ID-based ElGamal signature scheme in Section IV. Some extensions and variations of this generalized ElGamal signature scheme are also discussed in this section. The paper is concluded with a discussion of the proposed schemes in Section V.

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 ElGamal signature scheme 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. A cryptographic 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

Let p be a large prime and g be a generator of $\mathbb{Z}_p^*$. The user chooses $\alpha \in \mathbb{Z}_{p-1}$ as his private key and then computes $\beta = g^\alpha \bmod p$ as his public key. The parameters p, g, and β are public whereas the user keeps α secret. To sign a message, the user generates a random $k \in_R \mathbb{Z}_{p-1}$. Then he computes $r = g^k \bmod p$ and $s = k^{-1}(m - r\alpha) \bmod (p - 1)$. The (r, s) pair is the signature of message m. The equation

$$m \equiv \alpha r + ks \pmod{p-1} \tag{1}$$

called signature equation and verification is done by checking the congruence $g^m \stackrel{?}{\equiv} \beta^r r^s \bmod p$. Security of ElGamal signature relies on the discrete logarithm problem (DLP) since solving α from β or s from r, m, β can be reduced to solving DLP in $\mathbb{Z}_p^*$.

C. The Meta-ElGamal Signature Scheme

Horster et al. [6] showed that many variations of the basic ElGamal signature are possible by modifying the signature equation. Instead of using ElGamal's original signature equation, one can use the general equation

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

to obtain a signature, where (A, B, C) is a permutation of the parameters (m, r, s), q is a divisor of $p-1$, and g is an element in $\mathbb{Z}_p^*$ of order q. The signature can be verified by checking the equation:

$$g^A \stackrel{?}{\equiv} \beta^B r^C \pmod p \tag{2}$$

By these permutations six possible signatures can be obtained.

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

$$\pm A \equiv \pm \alpha B \pm kC \pmod q$$

where (A, B, C) is a permutation of the coefficients mentioned.

The generalization can be extended further by choosing A, B, C as general functions of m, r, s, instead of just products of two. The functions must be chosen carefully to guarantee the solvability and security. To guarantee solvability, it is necessary that the parameter s can be extracted from the equation. To guarantee security, the parameters m, r, s have to occur in at least one of the three coefficients. Also, the insecure rs and ms variants should be avoided.

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

An insecure ms variant occurs if (A, B, C) is a permutation of $(ms, r, 1)$: Assume that (r, s) is a valid signature observed by an adversary for some message m. For an arbitrary message m', the adversary computes s' as $s' = {m'}^{-1} ms$ and takes $r' = r$. Then (r', s') will be a valid signature for m'.

III. The Basic ID-based ElGamal Signature Scheme

An ID-based signature scheme consists of four algorithms: Setup, Extract, Sign, and Verify. In Setup, the trusted private key generator (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 by using the public parameters and the signer's identity.

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

- Setup: 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 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 $\langle G_1,\ G_2,\ e,\ P,\ P_{pub},\ H,\ H_1 \rangle$ where H and H_1 are secure hash functions.

- 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:

$$r = H(kP)$$
$$U = k^{-1}(mP - rS_{ID})$$

  The signature for the message m is (kP, U)

- Verify: Given ID, a message m, and a signature (kP, U), the signature is valid if the following equation holds.

$$e(U, kP)e(Q_{ID}, P_{pub})^r \stackrel{?}{=} e(P, P)^m \tag{3}$$

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

$$\begin{aligned} e(U, kP)e(Q_{ID}, P_{pub})^r &= e(k^{-1}(mP - rS_{ID}), kP)e(Q_{ID}, P_{pub})^r \\ &= e(mP - rS_{ID}, P)e(rS_{ID}, P) \\ &= e(mP, P) \\ &= e(P, P)^m \end{aligned}$$

The above scheme is the ID-based version of the original ElGamal signature scheme. The conversion process, which will also be used for other signature equations, is described below:

In the original ElGamal scheme, the signature equation is $m \equiv \alpha r + ks \bmod (p-1)$ where $r = g^k$ and the signature is (r, s). Since additive elliptic curve groups are used in ID-based structure, the signing equation and r will be slightly different. Signing equation for the ID-based ElGamal signature is:

$$mP = rS_{ID} + kU$$

Uppercase letters are used to denote elements of the elliptic curve group. S_{ID} is the private key of the user, so it is a natural replacement for α in the original scheme. U is a part of the signature and it is the replacement for s. We cannot use m directly since it is not a member of elliptic curve group; therefore mP is used to replace m. Here we can also use mQ_{ID} or mP_{pub} instead of mP and get a slightly different signature scheme.

A natural choice for r in the ID-based scheme is to compute r as $r = kP$ since r equals g^k in the original scheme. However, r must be an integer in $\mathbb{Z}_p$ in the signature equation, so we use a hash function and compute r as $r = H(kP)$. Additionally, since kP is needed for verification (3), the signature will be issued as (kP, U) instead of (r, U).

IV. The Generalized ID-based ElGamal Signature and its Variants

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

$$A = BS_{ID} + kC \tag{4}$$

where (A, B, C) is a permutation of the parameters (m, r, U), instead of the basic equation $mP = rS_{ID} + kU$. Note that, not all the permutations generate useful variants. We should consider that U is a member of elliptic curve group, and $m, r \in \mathbb{Z}_p$. Accordingly, A and C should be members of the elliptic curve group, but not B. Also note that, we can use mP and rP instead of m and r , in cases where they need to be members of the elliptic curve group.

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

$$mP = rS_{ID} + kU \tag{5}$$

$$U = rS_{ID} + kmP \tag{6}$$

$$U = mS_{ID} + krP \tag{7}$$

$$rP = mS_{ID} + kU \tag{8}$$

Note that, the two variants where U is a coefficient of S_{ID} do not produce useful signing equations.

In the variants where kP is not needed for verification, r can be computed as $e(P, P)^k$ and the signature for m will be (r, U). This has the advantage that we can get rid of one pairing operation in the verification phase. Additionally, since the signer knows k, he can compute $e(P, P)^k$ without any pairing computation. As can be seen in Table III, r is taken as $e(P, P)^k$ in (6) and (7). Note that, in (5) and (8), we need the value of kP for verification. In that case r will be computed as $r = H(kP)$ and the signature for m will be (kP, U). We can also compute r as $H(m, kP)$ instead of $H(kP)$ or $e(P, P)^k$. In that case, m does not need to occur in the signing equations.

We can generate more variants by using different permutations. Instead of choosing (A, B, C) as a permutation of (m, r, U), we can also choose them as a permutation of $(mr, U, 1)$, $(mr, mU, 1)$ and $(mr, rU, 1)$. Signs of A, B, and C can be changed by multiplying them by ± 1. We can also use a general function $f(m, r)$ instead of just product mr. Note that, unlike the original ElGamal variants, we cannot choose (A, B, C) as a permutation of $(mU, rU, 1)$, since we cannot extract U from the signing equation in these variants. The signature equations for these ID-based ElGamal variants can be found in Table I.

The verification equations and other details for all signatures are summarized in Table III. Group I lists the variants that are obtained by permuting (m, r, U) and $(1, r, U)$. Group II lists the variants that are obtained by permuting $(mr, U, 1)$. Group III lists the variants that are obtained by permuting $(mr, mU, 1)$. Group IV lists the variants that are obtained by permuting $(mr, rU, 1)$ and $(r, rU, 1)$. Group V shows the rU variants discussed in Section IV-A. Finally group VI shows the variants discussed in Section IV-B that were not possible on the basic ElGamal signatures.

No.	A	B	C	ElGamal Variant	ID-Based Signature
ID I.1	m	r	U	$m \equiv \alpha r + ks$	$mP = rS_{ID} + kU$
ID I.2	r	m	U	$r \equiv \alpha m + ks$	$rP = mS_{ID} + kU$
ID I.3	U	r	m	$s \equiv \alpha r + km$	$U = rS_{ID} + kmP$
ID I.4	U	m	r	$s \equiv \alpha m + kr$	$U = mS_{ID} + rkP$
ID II.1	1	mr	U	$1 \equiv mr\alpha + ks$	$P = mrS_{ID} + kU$
ID II.2	mr	1	U	$mr \equiv \alpha + ks$	$mrP = S_{ID} + kU$
ID II.3	U	mr	1	$s \equiv mr\alpha + k$	$U = mrS_{ID} + kP$
ID II.4	U	1	mr	$s \equiv \alpha + kmr$	$U = -S_{ID} - mrkP$
ID III.1	1	mr	mU	$1 \equiv mr\alpha + kms$	$P = mrS_{ID} + mkU$
ID III.2	mr	1	mU	$mr \equiv \alpha + kms$	$mrP = S_{ID} + kmU$
ID III.3	mU	mr	1	$ms \equiv mr\alpha + k$	$mU = mrS_{ID} + kP$
ID III.4	mU	1	mr	$ms \equiv \alpha + kmr$	$mU = S_{ID} + mrkP$
ID IV.1	mr	1	Ur	$mr \equiv \alpha + krs$	$mrP = S_{ID} + rkU$
ID IV.2	1	mr	Ur	$1 \equiv mr\alpha + krs$	$P = mrS_{ID} + rkU$
ID IV.3	Ur	1	mr	$rs \equiv \alpha + mrk$	$rU = S_{ID} + mrkP$
ID IV.4	Ur	mr	1	$rs \equiv mr\alpha + k$	$rU = mrS_{ID} + kP$

TABLE I
ElGamal variants and the corresponding ID-based ElGamal signature equations.

A. Security Analysis of Proposed Schemes

The generalized ElGamal signature schemes of Horster et al. [6] are believed to be secure except two insecure variants. The two insecure variants in the generalized ElGamal signature schemes are the rs and ms variants as discussed in Section II-C. The corresponding ID-based variants are the rU and mU variants. These variants occur if (A, B, C) is a permutation of $(rU, m, 1)$ or $(mU, r, 1)$, respectively.

The mU variants are completely insecure and the attack works similar to the attack for the ms variant of the basic ElGamal signature: Assume that the (r, U) pair is a valid signature observed by the adversary for message m. For an arbitrary message m', the adversary computes $U' = m'^{-1}mU$ and uses $r' = r$. Then (r', U') pair will be a valid signature for m'.

This is not always the case for the rU variants; the attack on the basic ElGamal rs variants does not work for two of the four ID-based rU variants. Signature and verification equation for the rU variants can be seen in Table II.

In Table II, the variants V.3, V.4 and V.6 are insecure. The attack for these rU variants works as follows: For an arbitrary message m, the adversary chooses $C \in_R G_1$. Then he substitutes $e(C, P)$ for $e(U, rP)$ in the verification equation and computes r. After that, he computes $U = r^{-1}C$. The (r, U) pair will be a valid signature for the message m.

The variants V.1, V.2 and V.5 in Table II seem to be secure since an attacker cannot extract r from the verification equation. Therefore, we have three more ID-based signatures from the rU variants.

	Signature equation	Verification equation
V.1	$mP = S_{ID} + rkU$	$e(U, kP)^r e(Q_{ID}, P_{pub}) = e(P, P)^m$
V.2	$P = mS_{ID} + rkU$	$e(U, kP)^r e(Q_{ID}, P_{pub})^m = e(P, P)$
V.3	$rU = -mS_{ID} + kP$	$e(U, rP)e(Q_{ID}, P_{pub})^m = r$
V.4	$rU = -S_{ID} + mkP$	$e(U, rP)e(Q_{ID}, P_{pub}) = r^m$
V.5	$P = S_{ID} + rkU$	$e(U, kP)^r e(Q_{ID}, P_{pub}) = e(P, P)$
V.6	$rU = -S_{ID} + kP$	$e(U, rP)e(Q_{ID}, P_{pub}) = r$

TABLE II
THE rU VARIANTS

B. Efficiency of the Proposed Schemes

Computing a signature requires a hash function evaluation or a pairing evaluation depending on how r is computed, as well as some additional computation in G_1. Several inversions modulo q may also be needed depending on the signature equation.

The cost of verifying a signature will be dominated by the pairing computations, which is the most expensive operation. Two or three pairing computations are needed to verify a signature depending on the signing equation. Note that, the value $e(P, P)$ is fixed, so it needs to be computed only once. Also the value $e(Q_{ID}, P_{pub})$ is fixed for a particular user, so it needs to be computed once for each user.

More efficient variants can be obtained by modifying the generalized signature equation (4) as

$$A = BS_{ID} + kCS_{ID} \tag{9}$$

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

By the help of bilinear pairings we can solve U from the signature equation (9) if we choose (A, B, C) as a permutation of (m, r, U), $(mr, U, 1)$ or $(m, rU, 1)$. Note that B and C cannot be a member of the elliptic curve group; hence U should be in A's position. So we get six more variants by using equation (9). These variants are:

$$\begin{aligned} U &= rS_{ID} + kmS_{ID} \\ U &= mS_{ID} + krS_{ID} \\ U &= rmS_{ID} + kS_{ID} \\ U &= S_{ID} + kmrS_{ID} \\ rU &= mS_{ID} + kS_{ID} \\ rU &= S_{ID} + kmS_{ID} \end{aligned}$$

The value of kQ_{ID} will be needed for verification. Therefore r is computed as $r = H(kQ_{ID})$ for these variants. For a message m the signature will be (kQ_{ID}, U). We can also compute r as $r = H(m, kQ_{ID})$ and remove m from the signing equations. Group VI of Table III shows the verification equations and other details for these schemes.

As observed by Barreto et al. [1], the number of pairing operations needed can be reduced further by changing the definitions of S_{ID} and Q_{ID} as

$$\begin{aligned} Q_{ID} &= (H_1(ID) + s)P, \\ S_{ID} &= (H_1(ID) + s)^{-1}P. \end{aligned}$$

For instance, for the signature (r, U), $r = e(P, P)^k$, $U = (k + mr)S_{ID}$, the verification equation becomes

$$r = e(U, Q_{ID})e(P, P)^{-mr},$$

and the number of pairing evaluations needed is reduced to one.

A similar modification can also be applied to the other signature schemes discussed in this paper to reduce the number of pairing evaluations in each verification.

C. Embedding Previously Known ID-based Signatures

Recently many ID-based signature schemes have been proposed. Most of these signatures [11], [8], [5], [3] can be seen as special instances of our generalized scheme:

- In Paterson's scheme [8], the signature (kP, U) is computed as

  $$\begin{aligned} r &= H(kP) \\ U &= k^{-1}(H_2(m)P + rS_{ID}) \end{aligned}$$

  where H_2 is a secure hash function. Paterson's scheme is equivalent to ID I.1 of Table III where a second hash function H_2 is used for message digest.
- In Cha-Cheon's scheme [3], the signature (kQ_{ID}, U) is computed as

  $$\begin{aligned} r &= H(m, kQ_{ID}) \\ U &= (r + k)S_{ID} \end{aligned}$$

  Cha-Cheon's scheme is the same as ID VI.7.
- In Yi's scheme [11], the signature (kP, U) is computed as

  $$\begin{aligned} r &= H(m, kP) \\ U &= kP_{pub} + rS_{ID} \end{aligned}$$

  Yi's scheme is equivalent to ID I.7, where, P_{pub} is used instead of P and the verification procedure is modified accordingly.
- In Hess's scheme [5], the signature (v, U) is computed as

  $$\begin{aligned} r &= e(P_1, P)^k \\ v &= H(m, r) \\ U &= kP_1 + vS_{ID} \end{aligned}$$

  where P_1 is an arbitrary point on the curve. Hess's scheme can be converted into ID II.3 with $P_1 = P$ and using mr instead of $v = H(m, r)$. Besides, in Hess's scheme, verification takes an extra step for checking $v \stackrel{?}{=} H(m, r)$.

No.	r	U	Signature	Verification
ID I.1	$r = H(kP)$	$U = k^{-1}(mP - rS_{ID})$	(kP, U)	$e(U, kP)e(Q_{ID}, P_{pub})^r = e(P, P)^m$
ID I.2	$r = H(kP)$	$U = k^{-1}(rP - mS_{ID})$	(kP, U)	$e(U, kP)e(Q_{ID}, P_{pub})^m = e(P, P)^r$
ID I.3	$r = e(P, P)^k$	$U = kmP - rS_{ID}$	(r, U)	$e(U, P)e(Q_{ID}, P_{pub})^r = r^m$
ID I.4	$r = e(P, P)^k$	$U = rkP - mS_{ID}$	(r, U)	$e(U, P)e(Q_{ID}, P_{pub})^m = r^r$
ID I.5	$r = H(m, kP)$	$U = k^{-1}(P - rS_{ID})$	(kP, U)	$e(U, kP)e(Q_{ID}, P_{pub})^r = e(P, P)$
ID I.6	$r = H(m, kP)$	$U = k^{-1}(rP - S_{ID})$	(kP, U)	$e(U, kP)e(Q_{ID}, P_{pub}) = e(P, P)^r$
ID I.7	$r = H(m, kP)$	$U = kP - rS_{ID}$	(kP, U)	$e(U, P)e(Q_{ID}, P_{pub})^r = e(P, kP)$
ID I.8	$r = H(m, kP)$	$U = rkP - S_{ID}$	(kP, U)	$e(U, P)e(Q_{ID}, P_{pub}) = e(P, kP)^r$
ID II.1	$r = H(kP)$	$U = k^{-1}(P - mrS_{ID})$	(kP, U)	$e(U, kP)e(Q_{ID}, P_{pub})^{mr} = e(P, P)$
ID II.2	$r = H(kP)$	$U = k^{-1}(-S_{ID} + mrP)$	(kP, U)	$e(U, kP)e(Q_{ID}, P_{pub}) = e(P, P)^{mr}$
ID II.3	$r = e(P, P)^k$	$U = kP - mrS_{ID}$	(r, U)	$e(U, P)e(Q_{ID}, P_{pub})^{mr} = r$
ID II.4	$r = e(P, P)^k$	$U = mrkP - S_{ID}$	(r, U)	$e(U, P)e(Q_{ID}, P_{pub}) = r^{mr}$
ID III.1	$r = H(kP)$	$U = k^{-1}(m^{-1}P - rS_{ID})$	(kP, U)	$e(U, kP)e(Q_{ID}, P_{pub})^r = e(P, P)^{m^{-1}}$
ID III.2	$r = H(kP)$	$U = k^{-1}(rP - m^{-1}S_{ID})$	(kP, U)	$e(U, kP)e(Q_{ID}, P_{pub})^{m^{-1}} = e(P, P)^r$
ID III.3	$r = e(P, P)^k$	$U = m^{-1}kP - rS_{ID}$	(r, U)	$e(U, P)e(Q_{ID}, P_{pub})^r = r^{m^{-1}}$
ID III.4	$r = e(P, P)^k$	$U = rkP - m^{-1}S_{ID}$	(r, U)	$e(U, P)e(Q_{ID}, P_{pub})^{m^{-1}} = r^r$
ID IV.1	$r = H(kP)$	$U = k^{-1}(mP - r^{-1}S_{ID})$	(kP, U)	$e(U, kP)e(Q_{ID}, P_{pub})^{r^{-1}} = e(P, P)^m$
ID IV.2	$r = H(kP)$	$U = k^{-1}(r^{-1}P - mS_{ID})$	(kP, U)	$e(U, kP)e(Q_{ID}, P_{pub})^m = e(P, P)^{r^{-1}}$
ID IV.3	$r = e(P, P)^k$	$U = mkP - r^{-1}S_{ID}$	(r, U)	$e(U, P)e(Q_{ID}, P_{pub})^{r^{-1}} = r^m$
ID IV.4	$r = e(P, P)^k$	$U = r^{-1}kP - mS_{ID}$	(r, U)	$e(U, P)e(Q_{ID}, P_{pub})^m = r^{r^{-1}}$
ID IV.5	$r = H(m, kP)$	$U = k^{-1}(P - r^{-1}S_{ID})$	(kP, U)	$e(U, kP)e(Q_{ID}, P_{pub})^{r^{-1}} = e(P, P)$
ID IV.6	$r = H(m, kP)$	$U = k^{-1}(r^{-1}P - S_{ID})$	(kP, U)	$e(U, kP)e(Q_{ID}, P_{pub}) = e(P, P)^{r^{-1}}$
ID IV.7	$r = H(m, kP)$	$U = kP - r^{-1}S_{ID}$	(kP, U)	$e(U, P)e(Q_{ID}, P_{pub})^{r^{-1}} = e(P, kP)$
ID IV.8	$r = H(m, kP)$	$U = r^{-1}kP - S_{ID}$	(kP, U)	$e(U, P)e(Q_{ID}, P_{pub}) = e(P, kP)^{r^{-1}}$
ID V.1	$r = H(kP)$	$U = k^{-1}r^{-1}(mP - S_{ID})$	(kP, U)	$e(U, kP)^r e(Q_{ID}, P_{pub}) = e(P, P)^m$
ID V.2	$r = H(kP)$	$U = k^{-1}r^{-1}(P - mS_{ID})$	(kP, U)	$e(U, kP)^r e(Q_{ID}, P_{pub})^m = e(P, P)$
ID V.3	$r = H(m, kP)$	$U = k^{-1}r^{-1}(P - S_{ID})$	(kP, U)	$e(U, kP)^r e(Q_{ID}, P_{pub}) = e(P, P)$
ID VI.1	$r = H(kQ_{ID})$	$U = (r + km)S_{ID}$	(kQ_{ID}, U)	$e(U, P) = e((r + km)Q_{ID}, P_{pub})$
ID VI.2	$r = H(kQ_{ID})$	$U = (m + kr)S_{ID}$	(kQ_{ID}, U)	$e(U, P) = e((kr + m)Q_{ID}, P_{pub})$
ID VI.3	$r = H(kQ_{ID})$	$U = (rm + k)S_{ID}$	(kQ_{ID}, U)	$e(U, P) = e((rm + k)Q_{ID}, P_{pub})$
ID VI.4	$r = H(kQ_{ID})$	$U = (1 + kmr)S_{ID}$	(kQ_{ID}, U)	$e(U, P) = e((1 + mkr)Q_{ID}, P_{pub})$
ID VI.5	$r = H(kQ_{ID})$	$U = r^{-1}(m + k)S_{ID}$	(kQ_{ID}, U)	$e(U, P)^r = e((m + k)Q_{ID}, P_{pub})$
ID VI.6	$r = H(kQ_{ID})$	$U = r^{-1}(1 + kmS_{ID})$	(kQ_{ID}, U)	$e(U, P)^r = e((mk + 1)Q_{ID}, P_{pub})$
ID VI.7	$r = H(m, kQ_{ID})$	$U = (r + k)S_{ID}$	(kQ_{ID}, U)	$e(U, P) = e((r + k)Q_{ID}, P_{pub})$
ID VI.8	$r = H(m, kQ_{ID})$	$U = r^{-1}(1 + k)S_{ID}$	(kQ_{ID}, U)	$e(U, P)^r = e((1 + k)Q_{ID}, P_{pub})$

TABLE III

THE GENERALIZED ID-BASED ELGAMAL SIGNATURES AND THEIR VERIFICATION EQUATIONS.

V. CONCLUSION

In this paper, converting the original ElGamal signature scheme into an ID-based signature scheme is investigated. We showed how the basic ID-based ElGamal signature scheme can be extended into a generalized ID-based signature scheme as in the work of Horster et al. on basic ElGamal signatures [6]. We discussed which variants are not possible and which variants are not secure in the ID-based setting. We also presented some original variants which were not possible on the basic ElGamal scheme.

Most of the ID-based signatures in the literature [11], [8], [5], [3] can be seen as special instances of the generalized ID-based signature scheme described in this paper. Therefore, our generalized scheme provides a unified framework for many of the previously proposed ID-based signatures. This framework also yields many new ID-based signature schemes that have not been explored before.

For future work, ways of proving the security of the proposed ID-based signature schemes can be investigated. One can also try to improve the efficiency of the proposed signature schemes by changing the signature and verification equations. The ideas presented in this paper can also be used to get new ID-based signatures with additional features such as message recovery and blinding.

REFERENCES

[1] 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.

[2] D. Boneh and M. Franklin. Identity-based encryption from the Weil pairing. In *Proc. of CRYPTO'01*, volume 2139 of *LNCS*, pages 213–229. Springer-Verlag, 2001.

[3] J. Cha and J.H. Cheon. An identity-based signature from gap diffie-hellman group. In *Proc. of PKC 2003*, volume 2567 of *LNCS*, pages 18–30. Springer-Verlag, 2003.

[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] F. Hess. Efficient identity based signature schemes based on pairings. In *Proc. of SAC'02*, volume 2595 of *LNCS*, pages 310–324. Springer-Verlag, 2003.

[6] P. Horster, H. Petersen, and M. Michels. Meta-elgamal signature schemes. In *Proc. of ACM Conference on Computer and Communications Security*, pages 96–107, 1994.

[7] A. Joux. A one round protocol for tripartite diffie-hellman. In *Proc. of ANTS-IV*, volume 1838 of *LNCS*, pages 385–394, 2000.
[8] K. Paterson. Id-based signatures from pairings on elliptic curves. Cryptology ePrint Archive, Report. http://eprint.iacr.org/2002/004.
[9] R. Sakai, K. Ohgishi, and M. Kasahara. Cryptosystems based on pairing. In *Proc. of SCIS'00*, 2003.
[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] X. Yi. An identity based signature scheme from the weil pairing. *IEEE Communication Letters*, 7(2):76–78, 2003.