

Identity Based KCDSA Signcryption

Jong-Ho Ryu, Youn-Seo Jeong, Dong-il Seo

Information Security Research Division, Electronics & Telecommunications Research Institute, 161
Gajeong-dong, Yuseong-gu, Daejeon, Korea

{ryubell, jys847, blusea}@etri.re.kr

Abstract - Many signcryption schemes have been proposed to provide authentication and confidentiality of a message efficiently. Among such a scheme, identity-based cryptography is one of the public key cryptography that does not require the certificate for a public key and pre-compute key pair. This scheme uses the public key that can be generated from an arbitrary identifier such as an email address, while the private key is derived by a trusted private key generator as occasion demands. In this paper, we propose identity-based KCDSA(Korean Certificate-based Digital Signature Algorithm) signcryption schemes providing the semantic security. Also, the proposed schemes support either the public verifiability or the forward secrecy. The proposed schemes are based on the standardized digital signature scheme and can be applied to the established KCDSA systems.

Keywords - KCDSA, Signcryption, Bilinear Pairing

1. Introduction

Authenticated encryption schemes in wireless security should provide authenticity and confidentiality of messages. One way to implement such schemes is first to sign a message and then to encrypt it, the other is vice versa. Recently, combined scheme of signaturing and encryption was proposed, called *signcryption* scheme. It was claimed that authenticity, confidentiality, integrity, non-repudiation and authentication was gained and the efficiency is superior.

The signcryption scheme was proposed to perform signature and encryption simultaneously by supporting the mentioned above more efficiently than the *sign-then-encrypt* scheme[3]. In order to provide the public verifiability and the forward secrecy, the signcryption scheme has been studied in the direction of the identity(ID)-based signcryption[4,13,14, 15,18].

An identity-based cryptography was introduced by Shamir [1]. The identity-based cryptography scheme has the ability to use any string, such as an email address or an IP address as a public key, while the corresponding private key can be derived by a trust key generator user's own identity information. Especially, the identity-based encryption scheme [10] uses bilinear map, $\hat{e}: \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$, over supersingular elliptic curves. This scheme performs encryption and decryption procedure for messages using weil pairing. And the security of this scheme is based on a natural analogue of the computational Diffie-Hellman assumption on elliptic curves.

John Malone-Lee proposed the first identity-based signcryption scheme[14]. The identity-based signcryption scheme [18] satisfies the semantic security under the hardness of the DBDH(Decisional Bilinear Diffie-Hellman) problem and the verification of the signcrypted message by the third party without private key. Also, these schemes reduced communication overhead and supported the forward secrecy.

Our Contribution: KCDSA was proposed for the signcryption [11]. In this paper, we propose the identity-based KCDSA signcryption schemes providing the semantic security under the Decisional Bilinear Diffie-Hellman assumption. The proposed signcryption schemes satisfy either the semantic security and the forward secrecy or the semantic security and the public verifiability, and the efficiency of the proposed schemes is equivalent to that of the *Libert & quisquater's* schemes.

Organization: Section 2 of this paper describes the principles and procedure of the existing identity-based signcryption, section 3 presents the KCDSA signcryption scheme based on the exponentiation complexity, section 4 presents the previous identity based signcryption scheme. In section 5, we propose identity-based KCDSA signcryption schemes which provide both the public verifiability and the forward secrecy.

2. ID-based Signcryption Scheme

2.1 Procedure of ID-based Signcryption

An identity-based signcryption scheme uses four steps which are the following ;

- **Setup.** Given a security parameter k , the private key generator generates the global system parameters.
- **Extract.** Given a string ID representing the identity of some party, the private key generator computes the corresponding private key d_{ID} for given ID .
- **Signcrypt.** Signcrypt by taking as input d_{ID_i} , ID_b and message m to obtain ciphertext .
- **Unsigncrypt.** Unsigncrypt takes as input d_{ID_s} , ID_a and a ciphertext σ to obtain a original message m or symbol \perp which indicates that the ciphertext was invalid.

2.2 Security of ID-based Signcryption

Malone-Lee defined the extended security notions for identity based signcryption schemes. Definition 1 defines the indistinguishability of encryption against adaptive chosen ciphertext attacks and Definition 2 does the unforgeability against adaptive chosen message attacks.

Definition 1. An identity-based signcryption scheme has the indistinguishability of encryption against adaptive chosen ciphertext attacks property (IND-ISC-CCA) if no polynomially bounded adversary has non-negligible advantage in the following game played by a challenger C and an adversary A .

- C takes a security parameter k and runs setup procedure to obtain the system parameters. It sends the system parameters to A .
- A may make the queries to C .
 - Key extraction queries: A requests the private key through the extraction queries as input ID_i , and receives the calculated private key d_{ID_i} from C .
 - Signcryption queries: A requests the ciphertext through signcryption queries as input ID_i, ID_j and a message M , and C calculates the private key $d_{ID} = Extract(ID_i)$ and $Unsigncrypt(M, d_{ID}, ID_j)$, and then sends the result to A .
 - Unsigncryption queries: A requests the result of unsigncryption procedure through the unsigncryption queries as input ID_i, ID_j and a ciphertext σ , and C calculates the private key $d_{ID} = Extract(ID_j)$ and $Unsigncrypt(\sigma, d_{ID}, ID_i)$, and then sends the result to A . This result can be the message M or symbol \perp which indicates that the ciphertext is invalid.
- C chooses two messages M_0 and M_1 , and two identities, ID_a and ID_b , on which it wishes to be challenged. A cannot request the private key corresponding to ID_a nor ID_b . C takes a bit b from $\{0,1\}$ and computes $d_{ID_a} = Extract(ID_a)$ and then computes $Signcrypt(M_b, d_{ID_a}, ID_b)$. C sends the result to A .
- A may make queries just like in the second step. However A cannot ask the key extraction queries ID_a nor ID_b , and the unsigncryption query $(ID_a, ID_b, ciphertext)$.
- A announces a bit b' and wins the game if $b' = b$.

The advantage of adversary A is defined to be

$$Adv(A) = |\Pr[b' = b] - 0.5|.$$

Definition 2. An identity-based signcryption scheme has the existential unforgeability under adaptive chosen message attacks property (EF-ISC-ACMA) if no polynomially bounded adversary has non-negligible advantage in the following game played by a challenger C and an adversary A .

- C takes a security parameter k and runs setup procedure to obtain the system parameters. It sends the system parameters to A .
- A performs a polynomially bounded number of requests just like in the previous definition 1.
- A computes a 3-tuple (σ^*, ID_a, ID_b) , where ID_a was not a key extract query, and wins the game if the result of $Unsigncrypt(\sigma^*, ID_a, ID_b)$ is not the symbol \perp .

The advantage of adversary A is defined to be

$$Adv(A) = \Pr[A \text{ wins}].$$

Definition 3. (Forward secrecy) If long-term private keys of one or more entities are compromised, the secrecy of previous session keys established by honest entities is not affected.

2.3 Properties of Bilinear Pairing

Let $(\mathbb{G}_1, +)$ and (\mathbb{G}_2, \times) be two cyclic groups of large prime order q . \mathbb{G}_1 is the additive group of points of a elliptic curve over \mathbb{F}_p , and \mathbb{G}_2 is the multiplicative group of points of elliptic curve over \mathbb{F}_{p^2} . If a map $\hat{e}: \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ satisfies the following properties, it is said to be bilinear pairing[10,14].

- *Bilinearity*: For all $P, Q, R \in \mathbb{G}_1$, $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$, $\hat{e}(P+Q, R) = \hat{e}(P, R)\hat{e}(Q, R)$, and $\hat{e}(P, Q+R) = \hat{e}(P, Q)\hat{e}(P, R)$.
- *Non-degeneracy*: There exists $P, Q \in \mathbb{G}_1$ such that $\hat{e}(P, Q) \neq 1$. And if $\hat{e}(P, Q) = 1$ then P is the point at infinity.
- *Computability*: There is an efficient algorithm to compute $\hat{e}(P, Q)$ for all $P, Q \in \mathbb{G}_1$, and P is the generator of \mathbb{G}_1 .

3. KCDSA Signature Schemes

KCDSA is Korean standard digital signature algorithm. In this section, we describe a signature and a verification operation of KCDSA introduced in [11].

3.1 KCDSA and Modified KCDSA

The parameters of KCDSA consist of both the public information (p, q, g) and the private keys x_a and x_b . Collision-free hash function is denoted by $hash: \{0,1\}^* \rightarrow \mathbb{Z}_q$. Let $\mathbb{G} = \langle g \rangle$ be a finite cyclic group of order q and $g \in \mathbb{G}$ be a generator of \mathbb{G} . KCDSA algorithm and the user parameters of KCDSA algorithm are denoted as follows:

- p and q : large primes, and q is a prime divisor of $p-1$.
- g : a generator with an order of q in $\mathbb{G}^F(p)$, that is, $g^q \equiv 1 \pmod{p}$.
- $cert_a$ (and $cert_b$): signer A (verifier B)'s certificate, the certificate includes user's public key.
- z_a (and z_b): a hash value of signer A (verifier B) certificate, where $z_a = hash(cert_a)$ and $z_b = hash(cert_b)$.
- x_a (and x_b): signer A (verifier B)'s private key, where $x_a, x_b \in \mathbb{Z}_q^*$.
- y_a (and y_b): signer A (verifier B)'s public verification key, where $y_a \equiv g^{x_a^{-1}} \pmod{p}$ and $y_b \equiv g^{x_b^{-1}} \pmod{p}$.
- $x_a^{-1}, x_b^{-1} (\in \mathbb{Z}_q^*)$: multiplicative inverse modulo q of the private key.
- $hash()$: a collision-free hash function with an output of length q .

Fig.1 shows the protocols for obtaining the signature of KCDSA. The user's identifier is indicated on the most top line in the figure, and their lists of inputs are shown in brackets on the next line. Signer A sends the signature (r, s) and a message m to verifier B , and then verifier B checks on the validity of the signature by $r \stackrel{?}{=} hash(k)$.

<i>Signer A</i> ($m, x_a, cert_a$) <i>Signature Generation</i>	<i>Verifier B</i> ($cert_a$) <i>Signature Verification</i>
$x \in_R \mathbb{Z}_q^*$ $k = g^x \bmod p$ $r = hash(k)$ $e = r \oplus hash(z_a, m) \bmod q$ $s = x_a(x - e) \bmod q$ $\sigma = (m, r, s)$	$\xrightarrow{\sigma}$ $e = r \oplus hash(z_a, m) \bmod q$ $k = y_a^s g^e \bmod p$ check. $r \stackrel{?}{=} hash(k)$

Fig 1. KCDSA algorithm

The signcryption schemes proposed in section 5.1 and 5.2 don't offer the forward secrecy in Definition 3. We applies the modified KCDSA algorithm [11] as Fig.2 for supporting the forward secrecy.

<i>Signature Generation</i>	<i>Signature Verification</i>
$x \in_R \mathbb{Z}_q^*$ $k = g^x \bmod p$ $r = hash(k)$ $e = r \oplus hash(z_a, m)$ $s = x_a(x - e) \bmod q$ $\sigma = (m, e, s)$	$\xrightarrow{\sigma}$ $r = e \oplus hash(z_a, m)$ $k = y_a^s g^e \bmod p$ check. $r \stackrel{?}{=} hash(k)$

Fig 2. Modified KCDSA algorithm

There are two modifications. That is, a signer and a verifier compute e by $e = r \oplus hash(z_a, m)$ instead of $e = r \oplus hash(z_a, m) \bmod q$, and a signer sends the 3-tuple (m, e, s) instead of (m, r, s) , for a verifier to recover k without message m . Note that a signer's sending (m, e, s) instead of (m, r, s) doesn't affect the security of the signature scheme because a verifier could obtain e from r .

3.2 KCDSA Signcryption Schemes

Yum and Lee proposed two signcryption schemes based on KCDSA, namely *Zheng's model* and *Bao & Deng's model* as shown in Fig.3 and Fig.4. These schemes are shown that they have the properties of the security such as unforgeability, non-repudiation, and confidentiality.

<i>Signcryption</i>	<i>Unsigncryption</i>
$x \in_R \mathbb{Z}_q^*$ $k = y_b^x \bmod p$ $r = hash(k)$ $k_1 = KDF(k)$ $c = E_{k_1}(m)$ $e = r \oplus hash(z_a, m)$ $s = x_a(x - e) \bmod q$ $\sigma = (c, e, s)$	$\xrightarrow{\sigma}$ $k = (y_a^s g^e)^{x^{-1}} \bmod p$ $r = hash(k)$ $k_1 = KDF(k)$ $m = D_{k_1}(c)$ check. $e \stackrel{?}{=} r \oplus hash(z_a, m)$

Fig 3. KCDSA signcryption scheme in *Zheng's model*

The KCDSA signcryption scheme based on *Bao & Deng's model* has the public verifiability, that is, any third party without a private key can verify the signature and cannot recover the original message from a ciphertext. If necessary, the verifier received a valid KCDSA signature may send to others, who wish to verify that it is originated with a signer. The validity of (m, r, s) can be verified by anyone who knows a signer's public key. However, in this scheme anyone who wishes to verify a message's origin must know the plaintext m recovered by a verifier. We propose the ID-based KCDSA signcryption scheme that the knowledge of the plaintext m is not required for the public verification of a message m .

<i>Signcryption</i>	<i>Unsigncryption</i>
$x \in_R \mathbb{Z}_q^*$ $k_1 = g^x \bmod p$ $k_2 = y_b^x \bmod p$ $r = hash(k_1)$ $k_3 = KDF(k_2)$ $c = E_{k_3}(m)$ $e = r \oplus hash(z_a, m)$ $s = x_a(x - e) \bmod q$ $\sigma = (c, e, s)$	$\xrightarrow{\sigma}$ $k_1 = (y_a^s g^e) \bmod p$ $k_2 = k_1^{x^{-1}} \bmod p$ $r = hash(k_1)$ $k_3 = KDF(k_2)$ $m = D_{k_3}(c)$ check. $e \stackrel{?}{=} r \oplus hash(z_a, m)$

Fig 4. KCDSA signcryption scheme in *Bao & Deng's model*

4. Previous ID-based Signcryption Schemes

4.1 Malone-Lee's ID-based Signcryption

<i>Setup</i>	<i>Extract</i>
$H_1: \{0,1\}^* \rightarrow \mathbb{G}^*$ $H_2: \{0,1\}^* \rightarrow \mathbb{Z}_q^*$ $H_3: \mathbb{Z}_q^* \rightarrow \{0,1\}^n$ $t \in_R \mathbb{Z}_q^*$ $P_{pub} = tP$ <i>paras</i> : $(\mathbb{G}_1, \mathbb{G}_2, n, \hat{e}, P, P_{pub}, H_1, H_2, H_3)$	$Q_{ID} = H_1(ID)$ $d_{ID} = t Q_{ID}$
<i>Signcrypt</i> (d_{ID}, ID_b, m)	<i>Unsigncrypt</i> (ID_a, d_{ID}, σ)
$Q_{ID_b} = H_1(ID_b)$ $x \in_R \mathbb{Z}_q^*$ $U = xP$ $r = H_2(U, m)$ $V = r d_{ID_a} + x P_{pub}$ $y = \hat{e}(x P_{pub}, Q_{ID_a})$ $k = H_3(y)$ $c = r \oplus m$ $\sigma = (c, U, V)$	$Q_{ID_a} = H_1(ID_a)$ $y = \hat{e}(d_{ID_a}, U)$ $k = H_3(y)$ $m = k \oplus c$ $r = H_2(U, m)$ if $\hat{e}(V, P) \neq \hat{e}(Q_{ID_a}, P_{pub})^r \hat{e}(U, P_{pub})$ Return \perp Return m

Fig 5. *Malone-Lee's ID-based signcryption scheme*

This scheme cannot support the semantic security. If the signature on the plaintext is opened, then any attackers can

verify the signature on plaintexts M_0 and M_1 produced during the game defined in Definition 1, because the plaintext is used to calculate the signature[18].

4.2 Libert & Quisquater's ID-based Signcryption

Libert & Quisquater's ID-based signcryption scheme illustrated in Fig.6 can offer the semantic security and the public verifiability simultaneously. It uses $r = H_3(c, k_1)$ instead of $r = H_3(m, k_1)$ to support the semantic security, and this modification doesn't affect the unforgeability of the algorithm. Also it doesn't require the knowledge of the recovered plaintext m for the public verification.

The signcryption scheme described in Fig.7 satisfies the forward secrecy, but can support neither the public verifiability nor the forward secrecy. That is, even if an attacker knows the signer's private key d_{ID} , he is unable to find out the plaintext without knowing r .

Setup	Extract
$H_1: \{0,1\}^* \rightarrow \mathbb{G}_1$ $H_2: \mathbb{G}_2 \rightarrow \{0,1\}^n$ $H_3: \{0,1\}^* \times \mathbb{G}_2 \rightarrow \mathbb{Z}_q$ $t \in_R \mathbb{Z}_q^*$ $P_{pub} = tP$ <i>paras</i> : $(\mathbb{G}_1, \mathbb{G}_2, n, \hat{e}, P, P_{pub}, H_1, H_2, H_3)$	$Q_{ID} = H_1(ID)$ $d_{ID} = tQ_{ID}$
<i>Signcrypt</i> (d_{ID}, ID_b, m)	<i>Unsigncrypt</i> (ID_a, d_{ID}, σ)
$Q_{ID_b} = H_1(ID_b)$ $x \in_R \mathbb{Z}_q^*$ $k_1 = \hat{e}(P, P_{pub})^x$ $k_2 = H_2(\hat{e}(P_{pub}, Q_{ID_b})^x)$ $c = E_{k_2}(m)$ $r = H_3(c, k_1)$ $S = xP_{pub} - rd_{ID_b}$ $\sigma = (c, r, S)$	$Q_{ID_a} = H_1(ID_a)$ $k_1 = \hat{e}(P, S) \hat{e}(P_{pub}, Q_{ID_a})^r$ $t = \hat{e}(S, Q_{ID_b}) \hat{e}(Q_{ID_b}, d_{ID_b})^r$ $k_2 = H_2(t)$ $m = D_{k_2}(c)$ <i>if</i> $r \neq H_3(c, k_1)$ <i>Return</i> \perp <i>Return</i> m

Fig 6. Libert & Quisquater's ID-based signcryption scheme

<i>Signcrypt</i> (d_{ID}, ID_b, m)	<i>Unsigncrypt</i> (ID_a, d_{ID}, σ)
$Q_{ID_b} = H_1(ID_b)$ $x \in_R \mathbb{Z}_q^*$ $(k_1, k_2) = H_2(\hat{e}(P_{pub}, Q_{ID_b})^x)$ $c = E_{k_2}(m)$ $r = H_3(c, k_1)$ $S = xP_{pub} - rd_{ID_b}$ $R = rQ_{ID_b}$ $\sigma = (c, R, S)$	$Q_{ID_a} = H_1(ID_a)$ $(k_1, k_2) = H_2(\hat{e}(S, Q_{ID_b}), \hat{e}(R, d_{ID_b}))$ $m = D_{k_2}(c)$ $r = H_3(c, k_1)$ <i>if</i> $R \neq rQ_{ID_b}$ <i>Return</i> \perp <i>Return</i> m

Fig 7. Libert & Quisquater's ID-based signcryption scheme with the forward secrecy

5. ID-based KCDSA Signcryption Schemes

Proposed identity(ID)-based KCDSA signcryption schemes are based on the public key produced from identity information such as an email address while the KCDSA signature scheme is based on a certification data.

5.1 ID-based KCDSA Signcryption Scheme in Zheng's Model

We propose the ID-based KCDSA signcryption scheme using the original KCDSA scheme except that the signer uses the ciphertext c to calculate the value e . This scheme supports the semantic security described in Section 4 because the ciphertext c is used to compute the signature. Although the signature on the plaintext is opened, any attackers cannot verify the signature on plaintexts M_0 and M_1 produced during the game defined in Definition 1.

A signer computes signcrypted message $\sigma = (c, r, S)$ using the trusted key generator's public key P_{pub} , the signer's private key d_{ID} and the verifier's public key Q_{ID} . A verifier computes the value e using the public information, i.e. r, Q_{ID} , and ciphertext c . Also he can obtain the message m and verify the signature using the private key d_{ID} , as follows.

Setup	Extract
$H_1: \{0,1\}^* \rightarrow \mathbb{G}_1$ $H_2: \mathbb{G}_2 \rightarrow \{0,1\}^n$ $H_3: \text{Key Derivation Function}$ $H_4: \mathbb{G}_1 \times \{0,1\}^* \rightarrow \{0,1\}^n$ $t \in_R \mathbb{Z}_q^*$ $P_{pub} = tP$ <i>paras</i> : $(\mathbb{G}_1, \mathbb{G}_2, n, \hat{e}, P, P_{pub}, H_1, H_2, H_3, H_4)$	$Q_{ID} = H_1(ID)$ $d_{ID} = tQ_{ID}$
<i>Signcrypt</i> (d_{ID}, ID_b, m)	<i>Unsigncrypt</i> (ID_a, d_{ID}, σ)
$Q_{ID_b} = H_1(ID_b)$ $x \in_R \mathbb{Z}_q^*$ $k_1 = \hat{e}(Q_{ID_b}, P_{pub})^x$ $r = H_2(k_1)$ $k_2 = H_3(k_1)$ $c = E_{k_2}(m)$ $e = (r \oplus H_4(Q_{ID_b}, c)) \bmod q$ $S = xP_{pub} - ed_{ID_b}$ $\sigma = (c, r, S)$	$Q_{ID_a} = H_1(ID_a)$ $e = (r \oplus H_4(Q_{ID_a}, c)) \bmod q$ $k_1 = \hat{e}(Q_{ID_b}, S) \hat{e}(d_{ID_b}, Q_{ID_a})^e$ $k_2 = H_3(k_1)$ $m = D_{k_2}(c)$ <i>if</i> $r \neq H_2(k_1)$ <i>Return</i> \perp <i>Return</i> m

Fig 8. ID-based KCDSA signcryption scheme in Zheng's Model

The consistency is easy to verify since the k_1 computed by a verifier is the same as a signer's one. The computed k_1 is used to recover the plaintext m and verify the signature on message.

$$\hat{e}(Q_{ID_b}, S) \hat{e}(d_{ID_b}, Q_{ID_a})^e =$$

$$\hat{e}(Q_{ID_a}, xP_{pub} - etQ_{ID_a}) \hat{e}(Q_{ID_a}, etQ_{ID_a}) = \hat{e}(Q_{ID_a}, P_{pub})^x$$

5.2 ID-based KCDSA Signcryption Scheme in Dao & Deng's Model

This scheme supports the semantic security and the public verifiability. The public verifiability means that the third party without a private key can verify the signcrypted message, but he cannot obtain the original message. The k_1 used to verify the signature is computed by only using the public information such as the system parameters P , the trusted key generator's public key P_{pub} , a signer's public key Q_{ID_a} , the signature data S , and the computed value e . However, the k_2 used to recover the plaintext is computed by using the verifier's private key d_{ID_a} . Also, the proposed signcryption scheme does not require the knowledge of the plaintext m for the public verification of a message.

$Signcrypt(d_{ID_a}, ID_b, m)$	$Unsigncrypt(ID_a, d_{ID_a}, \sigma)$
$Q_{ID_a} = H_1(ID_b)$ $x \in_R \mathbb{Z}_q^*$ $k_1 = \hat{e}(P, P_{pub})^x$ $k_2 = \hat{e}(P_{pub}, Q_{ID_b})^x$ $r = H_2(k_1)$ $k_3 = H_3(k_2)$ $c = E_{k_3}(m)$ $e = (r \oplus H_4(Q_{ID_a}, c))$ $S = xP_{pub} - ed_{ID_a}$ $\sigma = (c, r, S)$	$Q_{ID_a} = H_1(ID_a)$ $e = (r \oplus H_4(Q_{ID_a}, c)) \bmod q$ $k_1 = \hat{e}(P, S) \hat{e}(P_{pub}, Q_{ID_a})^e$ $k_2 = \hat{e}(S, Q_{ID_b}) \hat{e}(Q_{ID_a}, d_{ID_b})^e$ $\xrightarrow{\sigma}$ $k_3 = H_3(k_2)$ $m = D_{k_3}(c)$ <i>if</i> $r \neq H_2(k_1)$ <i>Return</i> \perp <i>Return</i> m

Fig 9. ID-based KCDSA signcryption scheme with the public verifiability

The consistency is easy to verify since the k_1 and the k_2 computed by a verifier are the same as a signer's values. The computed k_1 is used to verify the signature, and the k_2 is used to recover the plaintext m .

$$k_1 = \hat{e}(P, S) \hat{e}(P_{pub}, Q_{ID_a})^e = \hat{e}(P, xP_{pub} - etQ_{ID_a}) \hat{e}(P, etQ_{ID_a}) = \hat{e}(P, P_{pub})^x$$

$$k_2 = \hat{e}(S, Q_{ID_b}) \hat{e}(Q_{ID_a}, d_{ID_b})^e = \hat{e}(xP_{pub} - Q_{ID_a}, Q_{ID_b}) \hat{e}(etQ_{ID_a}, Q_{ID_b}) = \hat{e}(P_{pub}, Q_{ID_b})^x$$

5.3 ID-based KCDSA Signcryption Scheme with the Forward Secrecy

The signcryption schemes proposed in section 5.1 and 5.2 don't offer the forward secrecy because an attacker can find out the plaintext if he knows the signer's private key. We propose the ID-based KCDSA signcryption scheme using the modified KCDSA scheme except that the signer uses the ciphertext c to calculate the value e and sends $\sigma = (c, E, S)$ instead of $\sigma = (c, e, S)$ to verifier. This scheme satisfies the semantic security and the forward secrecy, but cannot support

the public verifiability.

The verifier receives the signcrypted message $\sigma = (c, E, S)$ from the signer, and then he obtains the key used to decrypt the ciphertext c using the value E and the verifier's private key d_{ID_a} . Also he computes the r and e , and then verifies the signature. In this scheme, even if an attacker knows the signer's private key d_{ID_a} , he is unable to find out the plaintext without knowing e .

$Signcrypt(d_{ID_a}, ID_b, m)$	$Unsigncrypt(ID_a, d_{ID_a}, \sigma)$
$Q_{ID_a} = H_1(ID_b)$ $x \in_R \mathbb{Z}_q^*$ $k_1 = \hat{e}(Q_{ID_a}, P_{pub})^x$ $r = H_2(k_1)$ $k_2 = H_3(k_1)$ $c = E_{k_2}(m)$ $e = (r \oplus H_4(Q_{ID_a}, c))$ $S = xP_{pub} - ed_{ID_a}$ $E = eQ_{ID_a}$ $\sigma = (c, E, S)$	$Q_{ID_a} = H_1(ID_a)$ $k_1 = \hat{e}(Q_{ID_a}, S) \hat{e}(d_{ID_a}, E)$ $k_2 = H_3(k_1)$ $m = D_{k_2}(c)$ $\xrightarrow{\sigma}$ $r = H_2(k_1)$ $e = (r \oplus H_4(Q_{ID_a}, c)) \bmod q$ <i>if</i> $E \neq eQ_{ID_a}$ <i>Return</i> \perp <i>Return</i> m

Fig 10. ID-based KCDSA signcryption scheme with forward secrecy

6. Security and Efficiency

6.1 Security

It is possible to use the method described in [10] to prove the IND-ISC-CCA security of the proposed signcryption scheme in the random oracle model. In the *Libert & Quisquater's* scheme described in Fig.6, the value of r that composes the signcrypted message is $H_3(c, k_1)$, but in the proposed schemes the value e included in the signcrypted message is $H_2(k_1) \oplus H_4(Q_{ID_a}, c)$. Finally, the proof of the IND-ISC-CCA security of the proposed schemes is equivalent to the proof of security of the *Libert & Quisquater's* scheme except that we may replace the list (c, k, r) used to simulate the signcryption oracle managed by the list (c, k, e) .

The proof of the EF-ISC-ACMA security of the proposed schemes in the oracle model is similar to the method of [15] except $e = H_2(k_1) \oplus H_4(Q_{ID_a}, c)$ instead of $r = H_3(c, k_1)$. This proof means that any attackers cannot forge the signcrypted message under the computational Diffie-Hellman assumption.

The proposed signcryption schemes support the semantic security, which means that although the signature on the plaintext is opened, any attackers cannot verify the signature on plaintexts M_0 and M_1 produced during the game defined in Definition 1.

6.2 Efficiency

We compare the efficiency of ID-based KCDSA signcryption schemes proposed in this paper with that of the *Malone-Lee's* scheme and the *Libert & Quisquater's* schemes. In Table 1,

LQ scheme I means the *Libert & Quisquater's* ID-based signcryption scheme and LQ scheme II does that with the forward secrecy. And proposed scheme I, II, and III mean ID-based KCDSA signcryption scheme in *Zheng's* model in Section 5.1, ID-based KCDSA signcryption scheme in *Dao & Deng's* model in Section 5.2, and ID-based KCDSA signcryption scheme with the forward secrecy in Section 5.3, respectively.

In terms of both the communication overhead and the computational requirement, the proposed scheme I is more compact than the *Malone-Lee's* scheme. Also the efficiency of the proposed schemes is equivalent to the *Libert & Quisquater's* schemes in terms of both the communication overhead and the computational requirement. However, our schemes can be applied to the established KCDSA systems. In Table 1, "1/4" means the number of operation in the "signcryption / unsigncryption" procedure. Also $|c|$, $|q|$ and $|\mathbb{G}_1|$ means the length of the ciphertext c , q and \mathbb{G}_1 respectively.

Table 1. Comparison of ID-based signature schemes

- Ⓐ E evaluation
- Ⓑ Multiplicative in \mathbb{G}_1
- Ⓒ Exponentiation in \mathbb{G}_2
- Ⓓ Communication overhead

	<i>Malone-Lee's</i> Scheme	LQ Scheme I	LQ Scheme II	Proposed Scheme I	Proposed Scheme II	Proposed Scheme III
Ⓐ	1/4	2/4	1/2	1/2	2/4	1/2
Ⓑ	3/0	2/0	3/1	2/0	2/0	3/1
Ⓒ	1/1	2/2	1/0	1/1	2/2	1/0
Ⓓ	$ c + 2 \mathbb{G}_1 $	$ c + q + 2 \mathbb{G}_1 $	$ c + 2 \mathbb{G}_1 $	$ c + q + \mathbb{G}_1 $	$ c + q + \mathbb{G}_1 $	$ c + 2 \mathbb{G}_1 $

7. Conclusion

In this paper, we propose the ID-based KCDSA signcryption schemes providing the semantic security under the Decisional Bilinear Diffie-Hellman assumption. This is a stronger assumption than the hardness of the computational Bilinear Diffie-Hellman assumption. The proposed signcryption schemes satisfy either the semantic security and the forward secrecy or the semantic security and the public verifiability. In terms of the communication overhead and the computational requirement, the efficiency of the proposed schemes is equivalent to that of the *Libert & quisquater's* schemes. The proposed ID-based signcryption schemes are based on the standardized digital signature scheme, i.e. KCDSA scheme. Due to such a characteristic, these ID-based signcryption schemes can be widely applied to the application using the KCDSA algorithm.

REFERENCES

[1] Adi Shamir, "Identity-based Cryptosystems and Signature Schemes", *Advanced in Cryptology: Proceedings of CRYPTO'84*, LNCS 196, pp. 47-53 (1985)

[2] Y. Zheng, "Improved Public Key Cryptosystems Secure against Chosen Ciphertext Attacks", *Technical Report 94-1*, University of Wollongong, Australia (1994)

[3] Y. Zheng, "Digital Signcryption or How to Achieve $\text{Cost}(\text{signature} \& \text{encryption}) \ll \text{cost}(\text{signature}) + \text{cost}(\text{encryption})$ ", *Advanced in Cryptology - CRYPTO'97*, LNCS 1294, pp. 165-179 (1997)

[4] F. Bao, R. H. Deng, "A Signcryption Scheme with Signature directly Verifiable by Public Key", *1st International Workshop on Practice and Theory in Public Key Cryptography - PKC'98*, LNCS 1431, pp.55-59 (1998)

[5] C. H. Lim, P. J. Lee, "A Study on the Proposed Korean Digital Signature Algorithm", *Advanced in Cryptology - ASIACRYPT '98*, LNCS 1514, pp 175-185 (1998)

[6] Y. Zheng, H. Imai, "Efficient Signcryption Schemes On Elliptic Curves", *IFIP/SEC'98*, Chapman & Hall (1998)

[7] E. Fujisaki, T. Okamoto, "Secure Integration of Asymmetric and Symmetric Encryption Schemes", *Advanced in Cryptology - CRYPTO'99*, LNCS 1666, pp. 537-553 (1999)

[8] Yi Mu and Vijay Varadharajan, "Distributed signcryption", *Progress in Cryptology - INDOCRYPT 2000*, LNCS 1977, pp. 155-164 (2000)

[9] M. Bellare, P. Rogaway, "Random Oracles are Practical: A Paradigm for Designing Efficient Protocols", *First Annual Conference on Computer and Communications Security - ACM*, Available at <http://www-cse.ucsd.edu/users/mihir> (1993)

[10] Dan Boneh and Matt Franklin, "Identity-based Encryption From the Weil Pairing", *Advanced in Cryptology - CRYPTO '01*, LNCS 2139, pp. 213-228 (2001)

[11] Dae Hyun Yum and Pil Joong Lee, "New Signcryption Schemes based on KCDSA", *Information Security and Cryptology - ICISC 2001: 4th International Conference*, LNCS 2288, pp. 305-317 (2001)

[12] J. H. Koo, H. J. Kim, I. R. Jeong, D. H. Lee, J. I. Lim, "Jointly Unsigncryptable Signcryption Scheme", *Preproceedings of International Workshop on Information Security Application : WISA 2001*, Vol 2, pp. 397-407 (2001)

[13] H. Y. Jung, D. H. Lee, J. I. Lim, K. S. Chang, "Signcryption Schemes with Forward Secrecy", *Preproceedings of International Workshop on Information Security Application : WISA 2001*, Vol 2 (2001)

[14] John Malone-Lee, "Identity-based Signcryption", Available at <http://eprint.iacr.org/098> (2002)

[15] Florian Hess, "Efficient Identity-based Signature Schemes based on pairings", *Selected Areas In Cryptography: 9th Annual International Workshop - SAC 2002*, LNCS 2595, pp 310-324 (2003)

[16] Junbum Shin, Kwangsu Lee, and Kyungah Shim, "New DSA-verifiable Signcryption Schemes", *Information Security and Cryptology - ICISC 2002: 5th International Conference*, LNCS 2587, pp. 35-47 (2002)

[17] Jee Hea An, Yevgeniy Dodis, and Tal Rabin, "On the Security of Joint Signature and Encryption", *Advanced in Cryptology - EUROCRYPT'02*, LNCS 2332, pp. 83-106 (2002)

[18] B. Libert, J. J. Quisquater, "New identity-based signcryption schemes from pairings", *IEEE Information Theory Workshop* (2003)