Security Analysis of Two Signcryption Schemes

Guilin Wang[1], Robert H. Deng[1], DongJin Kwak[2], and SangJae Moon[2]

[1] Institute for Infocomm Research (I^2R),
21 Heng Mui Keng Terrace, Singapore 119613.
{glwang, deng}@i2r.a-star.edu.sg
[2] Mobile Network Security Technology Research Center,
Kyungpook National Univ., Korea.
neverdid@m80.knu.ac.kr, sjmoon@knu.ac.kr

Abstract. Signcryption is a new cryptographic primitive that performs signing and encryption simultaneously, at a cost significantly lower than that required by the traditional signature-then-encryption approach. In this paper, we present a security analysis of two such schemes: the Huang-Chang convertible signcryption scheme [12], and the Kwak-Moon group signcryption scheme [13]. Our results show that both schemes are insecure. Specifically, the Huang-Chang scheme fails to provide confidentiality, while the Kwak-Moon scheme does not satisfy the properties of unforgeability, coalition-resistance, and traceability.

Keywords: signcryption, digital signature, encryption.

1 Introduction

Background. In the area of computer communications and electronic transactions, a very important concern is how to send data in a confidential and authenticated way. Usually, confidentiality of delivered data is provided by encryption algorithms, and authentication of messages is guaranteed by digital signatures. In the traditional paradigm, these two cryptographic operations are performed in the order of signature-then-encryption. Zheng [25, 26] first introduced an interesting notion called *signcryption* to provide confidentiality, unforgeability, and non-repudiation for the delivered data *simultaneously*. The motivation is to achieve significantly lower overheads on both aspects of computation and communications than that of the traditional signature-then-encryption paradigm.

Following Zheng's pioneering work, a number of new schemes and improvements have been proposed [3, 18, 24, 27, 1, 21, 6, 12–14], while literatures [22, 4, 1, 6] study the formal models and security proofs for signcryption schemes. Originally, signcryption is performed by a sender Alice for a designated receiver Bob. In [26], a variant is proposed to support multiple designated receivers. Noticed that the non-repudiation protocols in [26] are inefficient since they are based on interactive zero-knowledge proofs, Bao and Deng [3] presented schemes so that a designated receiver can efficiently convert a signcrypted message into a publicly verifiable signature. Based on the same idea, Yum and Lee [24], and Shin et al. [21] proposed efficient schemes based on KCDSA and DSA [9]. In this paper, we call such schemes *convertible signcryptions*. In addition, Wang et al. [23] identified an interesting attack on a signcryption scheme proposed in [15]. Their attack allows a dishonest receiver Bob to forge a valid signcrypted message as if it were generated by Alice, under the assumption that Bob knows Alice's public key when he registers his public key. Furthermore, a newly convertible scheme based on the Schnorr signature scheme is presented in [23].

In [13], Kwak and Moon introduced a new notion called *group signcryption* by combining the concepts of group signature [8, 7, 2] and signcryption [25, 26] together. In such a scheme, a member Alice from a sending group G_A can produce a signcrypted message for the receiving group G_B so that any member of G_B can unsigncrypt such a ciphertext and then know this ciphertext must be generated by some member of G_A, but cannot identify who is the actual signer. In the event of dispute, however, as in group signatures, the group manager GM_A of G_A can open a valid signcrypted message and then reveal the identity of the true signer. To construct such a concrete scheme, Kwak and Moon first modified Mu et al.'s distributed schemes [17, 18] to obtain a distributed signcryption scheme supporting the confidentiality of the sender's ID. Then, based on this distributed signcryption scheme, they developed a concrete group signcryption scheme.

In the following, we introduce the security requirements for the convertible signcryption schemes and group signcryption schemes informally.

Convertible Signcryption. A convertible signcryption scheme should satisfy the following security requirements [3, 12]:

- Unforgeability: Except Alice, any attacker (including Bob) cannot forge a valid signcrypted message so that the verification equation is satisfied.
- Confidentiality: Except the designated receiver Bob, any third party cannot derive the plaintext from the signcrypted message.
- Non-repudiation: Once Alice generated a valid signcryption message, she cannot deny this fact. In other words, Bob can prove (maybe inefficiently) to a third party that such a signcrypted message is indeed generated by Alice.
- Convertibility: For any signcrypted message for receiver Bob, he can efficiently convert it into a publicly verifiable signature.

Note that those security requirements are almost the same as in standard signcryption schemes [25, 26], except the convertibility.

Group Signcryption. As the combination of group signatures [8, 7, 2] and signcryptions [25], a *secure* group signcryption scheme must satisfy the following security requirements [13]:

- Correctness: The signcrypted message produced by a group member must be accepted by the unsigncryption procedure.
- Unforgeability: Only valid group members are able to signcrypt a message on behalf of the group.
- Anonymity: With a valid decrypted message, identifying the individual who signcrypted the message is computationally hard for anyone but the group manager.

– Unlinkability: Deciding whether two valid unsigncrypted messages were generated by the same group member is computationally hard for anyone but the group manager.
– Exculpability: Neither a group member nor the group manager can signcrypt on behalf of other group members.
– Traceability: For any valid unsigncrypted message, the group manager can open it and find the true signer.
– Coalition-resistance: This means that a colluding subset of group members cannot generate a valid signcryption so that the group manager is unable to link it to one of the colluding group members.
– Confidentiality: Except the members belonging to the receiving group, any other party cannot derive the unsigncrypted message from the signcrypted message.

Our Work. In this paper, we present a security analysis of the Huang-Chang convertible signcryption scheme [12], and the Kwak-Moon group signcryption scheme [13]. Note that authenticated encryption does not necessarily provide the property of non-repudiation, so we call Huang-Chang scheme as convertible signcryption scheme, instead of convertible authenticated scheme. Our results show that both schemes do not meet all the desired security requirements. More Specifically, the Huang-Chang fails to provide confidentiality, while the Kwak-Moon scheme does not satisfy the properties of unforgeability, coalition-resistance, and traceability. In our analysis, we not only demonstrate concrete attacks to show the insecurity of those two schemes, but also discuss the reasons leading to such security flaws.

Organization. For self-contained, we first briefly review Zheng's original signcryption schemes in Section 2. Then, we review and analyze the Huang-Chang scheme and the Kwak-Moon scheme in Sections 3 and 4, respectively. Finally, Section 5 concludes the paper and proposes some future work.

2 Review of Zheng's Signcryption Schemes

In Zheng's two original signcryption schemes shown below, Alice signcrypts a message m and Bob unsigncrypts the ciphertext (c, r, s). Here, $(x_a, y_a = g^{x_a} \bmod p)$ and $(x_b, y_b = g^{x_b} \bmod p)$ denote the certified key pairs of Alice and Bob, respectively; $H(\cdot)$ is a strong one-way hash function; $H_k(\cdot)$ a keyed one-way hash function with key k; and (E_k, D_k) a pair of symmetric encryption/decryption algorithms. Note that Zheng's schemes are based on the Digital Signature Standard (DSS) [9], but with a minor modification to make his schemes more efficient. The two modified versions of DSS are referred to as SDSS1 and SDSS2, according to [25]. For more discussions on the security and efficiency of Zheng's schemes, please refer to [25, 26, 4].

Alice
choose $z \in_R Z_q$
compute $k = y_b^z \bmod p$
split k into k_1 and k_2
compute $r = H_{k_2}(m)$
$s = z(r + x_a)^{-1} \bmod q$ if SDSS1
$s = z(1 + x_a \cdot r)^{-1} \bmod q$ if SDSS2
$c = E_{k_1}(m)$

$\longrightarrow (c, r, s) \longrightarrow$

Bob
$k = (y_a \cdot g^r)^{s \cdot x_b} \bmod p$ if SDSS1
$k = (y_a^r \cdot g)^{s \cdot x_b} \bmod p$ if SDSS2
split k into k_1 and k_2
compute $m = D_{k_1}(c)$
verify $r \equiv H_{k_2}(m)$

3 The Huang-Chang Scheme and Its Security

3.1 Review of the Huang-Chang Scheme

The Huang-Chang scheme [12] is a combination of the the ElGamal encryption system [10] and the Schnorr signature scheme [20]. There are four phases in their scheme: setup, signcryption, unsigncryption and conversion. In the setup phase, system parameters are set. At the same time, a sender Alice and a receiver Bob register their public keys with a certificate authority (CA). In the signcryption phase, the signer Alice sincrypts a message for a specified receiver Bob. Using the unsigncrption algorithm, Bob checks whether an alleged ciphertext is generated by Alice. In the event of dispute, by using the conversion algorithm, Bob converts a valid ciphertext into a publicly verifiable signature to convince a judge (or any third party) that the ciphertex is indeed generated by Alice.

(1) Setup. Initially, the system parameters (p, q, g) are set, where p and q are two large primes satisfying $q|(p-1)$, and $g \in \mathbb{Z}_p^*$ is an element of order q. It is assumed that the discrete logarithm (DL) problem and computational Diffie-Hellman (CDH) problem are difficult in the multiplicative subgroup $G_q = \langle g \rangle$. At the same time, a publicly known one-way hash function $H(\cdot)$ is selected. In addition, each user i in the system picks a random number $x_i \in_R \mathbb{Z}_q$ as its private key, and then registers the corresponding public key $y_i = g^{x_i} \bmod p$ with the CA. In the following, we use subscripts a and b to denote the sender Alice and the receiver Bob, respectively. For example, (x_a, y_a) and (x_b, y_b) are the key pairs of Alice and Bob, respectively.

(2) Signcryption. To signcrypt a message $m \in \mathbb{Z}_p$ for the receiver Bob, the sender Alice does the following using her private key x_a.

(2.1) Pick a random number $k \in_R \mathbb{Z}_p^*$, and compute $c = m \cdot y_b^{-k} \bmod p$.
(2.2) Compute $r = H(m, y_b, g^k \bmod p) \bmod q$, and $s = k - x_a r \bmod q$.
(2.3) Finally, send the ciphertext (c, r, s) to the receiver Bob.

(3) Unsigncryption. Upon receiving the ciphertext (c, r, s), the receiver Bob uses his private key x_b to recover message m and check its validity as follows.

(3.1) Recover the message m by

$$m = c \cdot (y_a^r \cdot g^s)^{x_b} \bmod p. \tag{1}$$

(3.2) Accept the ciphertext (c, r, s) iff the following equality holds:

$$r \equiv H(m, y_b, y_a^r g^s \bmod p) \bmod q. \tag{2}$$

(4) Conversion. In later potential disputes, Bob just needs to reveal the message m and the corresponding signature (r, s). Then, a judge (or any third party) can check whether the triple (m, r, s) satisfies equation (2). If the answer is positive, it is concluded that Alice indeed generated the signature (r, s) for Bob.

3.2 The Secuity of the Huang-Chang Scheme

Obviously, the Huang-Chang scheme is indeed the combination of the ElGamal encryption algorithm and the Schnorr signature scheme. At the same time, it is widely believed that the ElGamal cryptosytem is secure in practice. Furthermore, the security of the Schnorr signature scheme is proved to be equivalent to the DL problem [19]. Based on the above observations, Huang and Chang provided elaborate but informal analysis to show that their scheme is also secure. Actually, they claimed that their scheme satisfies the following three security requirements:

(1) Unforgeability: Except Alice, any attacker (including Bob) cannot forge a valid ciphertext (c, r, s) for any message m so that the verification equations (1) and (2) are satisfied.
(2) Confidentiality: Except the designated receiver Bob, any third party cannot derive the message m from the ciphertext (c, r, s).
(3) Non-repudiation: Once Bob reveals a triple (m, r, s), anybody can verify that (r, s) is Alice's signature. Therefore, a judge can settle a possible dispute between Alice and Bob.

We note that the Huang-Chang scheme indeed satisfies the unforgeability and non-repudiation requirements. The reason is that if an adaptive attacker (including Bob) can forge a valid ciphertext triple (c, r, s) for a new message m so that both equations (1) and (2) hold, this exactly means the attacker has forged a standard Schnorr signature (r, s) for the message $m||y_b$. The latter is contrary to the known result that the Schnorr signature is *existentially unforgeable* [11] in the random oracle model [5], which is proved by Pointcheval and Stern in [19].

The correctness of their conclusion on the confidentiality is another story. Firstly, let $y_{ab} = g^{x_a \cdot x_b} \bmod p$, then equation (1) can be re-written as

$$m = c \cdot y_{ab}^r \cdot y_b^s \bmod p. \tag{3}$$

This equation implies that if the value y_{ab} is known, the plaintext m can be derived from ciphertext (c, r, s) and Bob's public keys y_b directly. So, the value of y_{ab} plays a pivotal role in the Huang-Chang scheme. Any party other than Alice and Bob cannot compute the value of y_{ab} from y_a and y_b, since it is assumed that the CDH assumption hold in the subgroup $G_q = \langle g \rangle$. However, the point is that equation (3) also means the value of y_{ab} can be carried out from a valid ciphertext (c, r, s) by the following equation:

$$y_{ab} = (m \cdot c^{-1} \cdot y_b^{-s})^{r^{-1}} \bmod p. \quad (4)$$

Therefore, if an eavesdropper obtains a valid ciphertext (c, r, s) for a message m, he or she can compute the value of y_{ab} from equation (4). Then, when a new valid ciphertext (c', r', s') is received or intercepted, the eavesdropper can decrypt it easily by computing $m' = c' \cdot y_{ab}^{r'} \cdot y_b^{s'} \bmod p$. In other words, the Huang-Chang scheme is vulnerable to the known-plaintext attack. Consequently, the security requirement of confidentiality is not guaranteed.

To sincrypt a large message m, i.e., $m \geq p$, the authors of [12] also proposed a variant of the above scheme called *convertible authenticated encryption scheme with message linkage*. The above attack applies to this variant, too. Specifically, one can get the value of y_{ab} from a known message-ciphertext pair. Then, using y_{ab} any new ciphertext can be decrypted easily by first computing the hidden random number $t = c \cdot y_{ab}^r \cdot y_b^s \bmod p$, and then recovering each block of the plaintext one by one. For more details, please check Section 3.1 of [12].

4 The Kwak-Moon Scheme and Its Security

4.1 Review of the Kwak-Moon Scheme

Similar to group signatures, the Kwak-Moon group signcryption scheme consists of five procedures: setup, join, signcryption, unsigncryption, and open. In the setup procedure, system parameters are set, while the join procedure allows each system user to register with the corresponding group manager and then get his/her group membership certificate. Then, using this group membership certificate one user can generate signcrypted messages on behalf of the group according to unsigncryption procedures, and sends it to the members in the receiving group. In unsigncryption procedures, users verify signcrypted messages originated from the sending group. By using the open procedure, the sending group manager can find out the identity of the true signer who issued a valid signcrypted messages on behalf of the sending group.

(1) Setup. To setup a group, the group manager GM_A performs as follows:

(1.1) Set group manager GM_A's RSA signature public key (n_A, e_A) and private key d_A, where the RSA modulus n_A is the product of two random primes with approximately equal length, and (e_A, d_A) satisfies $e_A \cdot d_A = 1 \bmod \phi(n_A)$.

(1.2) Select a discrete logarithm triple (p, q, g), where p and q are two large primes such $q|(p-1)$, and $g \in \mathbb{Z}_p^*$ is a generator of order q, such that the DL assumption and CDH assumption hold in the multiplicative subgroup $G_q = \langle g \rangle$. In addition, select a publicly known one-way hash function $H(\cdot)$ and a random element $h \in_R \mathbb{Z}_p^*$.

(1.3) The group manager GM_A keeps d_A as his secret key, and publishes $(p, q, g, h, H(\cdot), n_A, e_A)$ as the system parameters.

(2) Join. When a user l wants to join a group, the following interactive protocols is executed.

(2.1) User l who wants to join the group G_A generates his/her own group private key ϵ_l, and computes $\tau_l = h^{\epsilon_l} \bmod p$ as *group membership key*. Then he transfers τ_l to the group manager GM_A through secure channel and proves to group manager GM_A that he knows the discrete logarithm of τ_l to the base h. ϵ_l should be kept secret by the user l.
(2.2) Then, group manager GM_A calculates $v_l = \tau_l^{d_A} \bmod n_A$ as user l's membership certificate as in [7].
(2.3) When n registration applications from n users are received, group manager GM_A computes the following polynomial $f(x)$'s coefficients α_i, $i = 1, \cdots, n$:

$$f(x) = \prod_{i=1}^{n}(x - \tau_i) = \sum_{i=0}^{n} \alpha_i x^i \in \mathbb{Z}_q[x]. \quad (5)$$

Using the set $\{\alpha_0, \alpha_1, ..., \alpha_n\}$, a new set $\{\alpha'_0, \alpha'_1, \cdots, \alpha'_n\}$ is defined, where $\alpha'_0 = \alpha_0, \alpha'_n = \alpha_n, \alpha'_1 = \cdots = \alpha'_{n-1} = \sum_{i=1}^{n-1} \alpha_i \bmod q$. Let $\beta_i = g^{\alpha'_i} \bmod p$ for each $i = 1, \cdots, n$, and $A_l = \sum_{i=1,j=1,i\neq j}^{n-1} \alpha_j \tau_l^i \bmod q$ for each $l = 1, \cdots, n$. Then, each τ_l satisfies the following property:

$$F'(\tau_l) = g^{-A_l} \prod_{i=0}^{n} \beta_i^{\tau_l^i} = g^{-A_l} g^{\sum_{i=0}^{n} \alpha'_i \tau_l^i} = g^{f(\tau_l)} = 1 \bmod p. \quad (6)$$

(2.4) In order to create a group public key, group manager GM_A picks a random number $\gamma \in_R \mathbb{Z}_q^*$, and sets $\rho_l = -\gamma \cdot A_l \bmod q$ for user l. The *group public key* is defined as $\{\beta_0, ..., \beta_{n+1}\}$, where $\beta_{n+1} = g^{\gamma^{-1}} \bmod p$.
(2.5) Finally, the pair (v_l, ρ_l) is sent to group member l, while the group manager keeps γ, and all $\{\alpha_i\}$, $\{\tau_l\}$ secret.

(3) Signcryption. Now we assume that two groups, G_A and G_B, are set up according to the above procedures, and that the sender Alice belongs to G_A and the receiver Bob belongs to G_B. In order to signcrypt a message m for group G_B, Alice with her signing key $(\epsilon_a, \tau_a, v_a)$ performs as follows.

(3.1) Choose two random numbers $z, t \in_R \mathbb{Z}_q$, and compute $k = g^z \bmod p$.
(3.2) Split k into k_1 and k_2 with appropriate lengths.
(3.3) Evaluate $r = H_{k_2}(m)$.
(3.4) Set $s = z(r + \epsilon_a \cdot t)^{-1} \bmod q$ if SDSS1, or $s = z(1 + \epsilon_a \cdot r \cdot t)^{-1} \bmod q$ if SDSS2.
(3.5) Evaluate $w = H(m)$.
(3.6) Compute $\lambda_a = (t^{e_A} \cdot \tau_a \bmod n_A) \bmod q$, $\delta_a = g^{\epsilon_a t} \bmod p$, and $\theta_a = t \cdot v_a \bmod n_A$.

(3.7) The signcrypted message (c_1, c_2) is computed by

$$c_1 \leftarrow \{a_0, ..., a_{n+2}\} \leftarrow \{k\beta_0^{w\tau_a}, \beta_1^{w\tau_a}, ..., \beta_{n+1}^{w\tau_a}, g^{\lambda_a}\},$$
$$c_2 = E_{k_1}(ID_{G_A}||m||r||s||\delta_a||\theta_a),$$

where ID_{G_A} is the identity of group G_A that includes GM_A's public key (n_A, e_A).

(4) Unsigncryption. With the secret information (τ_b, ρ_b), Bob (or any member of G_B) can unsigncrypt the signcrypted message (c_1, c_2) as follows.

(4.1) Recover the secret session key k by

$$k = a_0(\prod_{i=1}^{n} a_i^{\tau_b^i})a_{n+1}^{\rho_b} = g^z \prod_{i=0}^{n} g^{w\tau_a\alpha_i\tau_b^i} = g^z(g^{f(\tau_b)})^{w\tau_a} = g^z \bmod p. \quad (7)$$

(4.2) Split k into k_1 and k_2.
(4.3) Decrypt $D_{k_1}(c_2) = ID_{G_A}||m||r||s||\delta_a||\theta_a$.
(4.4) Compute $\lambda'_a = ({\theta_a}^{e_A} \bmod n_A) \bmod q$.
(4.5) Accept (c_1, c_2) iff $r \equiv H_{k_2}(m)$, $k \equiv (\delta_a \cdot g^r)^s \bmod p$ if SDSS1 or $k \equiv (g \cdot \delta_a^r)^s \bmod p$ if SDSS2, and $a_{n+2} \equiv g^{\lambda'_a} \bmod p$.

(5) Open. In case of disputes, Bob forwards the (c_1, w) to group G_A's manager GM_A. Then, only the group manager GM_A can find the group member, Alice, who issued this signcryption. To do so, GM_A searches which τ_l belonging to G_A satisfying $a_i = (\beta_i^w)^{\tau_l}$, for all $i = 1, \cdots, n+1$.

4.2 The Secuity of the Kwak-Moon Scheme

The authors of [13] analyzed their scheme on both aspects of security and efficiency, and claimed that as the combination of group signatures [8, 7, 2] and signcryptions [25], their scheme satisfies all security requirements for group signcryption scheme listed in Section 1. However, we find this is not the fact. We now demonstrate two attacks to show that the Kwak-Moon scheme *does not* satisfy the following security requirements: coalition-resistance, traceability, and unforgeability.

Untraceability. In [13], it is argued that each v_l is the group manager's RSA signature for member l's group membership key τ_l and is sent to member l securely. So, no colluding subset can generate a valid correlated $(\epsilon_i, \tau_i, v_i)$ without the help of the right member and the group manager. This conclusion is incorrect. Firstly, after a careful checking the signcryption procedure we know that to generate a signcrypted message on behalf of the group G_A, it is sufficient that if one possesses a triple (ϵ, τ, v) such that the following equations are satisfied:

$$\tau = h^{\epsilon} \bmod p, \quad \text{and} \quad v = \tau^{d_A} \bmod n_A. \quad (8)$$

Therefore, a group member, say Alice, can forge a new triple $(\epsilon'_a, \tau'_a, v'_a)$ from her old triple $(\epsilon_a, \tau_a, v_a)$ by first selecting a random number ϵ, and then computing $(\epsilon'_a, \tau'_a, v'_a)$ as

$$\epsilon'_a = \epsilon_a \cdot \epsilon \bmod q, \quad \tau'_a = \tau_a^{\epsilon} \bmod p, \quad \text{and} \quad v'_a = v_a^{\epsilon} \bmod n_A. \tag{9}$$

It is easy to know that the resulting new triple $(\epsilon'_a, \tau'_a, v'_a)$ satisfies equations in (8). Consequently, Alice can use it to generate valid but untraceable signcrypted messages. That is, any member from receiving group will accept all signcrypted messages generated by using $(\epsilon'_a, \tau'_a, v'_a)$, according to signcryption procedure. When such signcrypted messages are presented, however, the group manager GM_A cannot identify the true singer, since Alice does not use her true certificate. This attack implies that the property of coalition-resistance should be proved rigorously.

Forgeability. In the following, we show that even with out any membership certificate, an attacker can also forge signcrypted messages on behalf of the sending group G_A. In other words, the Kwak-Moon scheme is universally forgeable. The authors of [13] argued that their scheme is unforgeable, since the keyed hash function $H_k(\cdot)$ behaves as a random function, and the group member's private key ϵ_a is not revealed to anyone. However, such argument does not guarantee the unforgeability. The basic idea of the following attack is to select random values for ϵ, θ, and τ, but computing λ and δ as the desired values. To forge a sincrypted message on behalf of group G_A, an outsider without any system secret can mount the following attack.

(1) Choose random numbers $\epsilon, z, t \in_R \mathbb{Z}_q$, and compute $k = g^z \bmod p$.
(2) Split k into k_1 and k_2 with appropriate lengths.
(3) Evaluate $r = H_{k_2}(m)$.
(4) Set $s = z(r + \epsilon \cdot t)^{-1} \bmod q$ if SDSS1, or $s = z(1 + \epsilon \cdot r \cdot t)^{-1} \bmod q$ if SDSS2.
(5) Evaluate $w = H(m)$.
(6) Select random number $\theta \in_R \mathbb{Z}_{n_A}$, and compute $\lambda = (\theta^{e_A} \bmod n_A) \bmod q$, $\delta = g^{\epsilon t} \bmod p$.
(7) Pick a random number $\tau \in_R \mathbb{Z}_p$, the signcrypted message (c_1, c_2) is computed by

$$c_1 \leftarrow \{a_0, \cdots, a_{n+2}\} \leftarrow \{k\beta_0^{w\tau}, \beta_1^{w\tau}, \cdots, \beta_{n+1}^{w\tau}, g^{\lambda}\},$$
$$c_2 = E_{k_1}(ID_{G_A}||m||r||s||\delta||\theta).$$

We explain our attack is successful. Firstly, note that equation (7) holds for the above forged ciphertext (c_1, c_2), since this is due to the property of the values (τ_b, ρ_b). This means any member of the receiving group, say Bob, can recover the secret session key k. Then, he can decrypt c_2 and get the values of $(ID_{G_A}, m, r, s, \delta, \theta)$. By computing $\lambda' = (\theta^{e_A} \bmod n_A) \bmod q$ $(= \lambda)$, Bob will find that $r \equiv H_{k_2}(m)$, $k \equiv (\delta \cdot g^r)^s \bmod p$ if SDSS1 or $k \equiv (g \cdot \delta^r)^s \bmod p$ if SDSS2, and $a_{n+2} \equiv g^{\lambda'} \bmod p$. This is, Bob will accepts such forged pair (c_1, c_2) as valid signcrypted messages. This attack results from the fact that the relationships among components of a group membership certificate are not fully used in sincryption procedure. In other words, to signcrypt a message in the Kwak-Moon scheme it is not necessarily to have a group membership certificate.

5 Conclusion

In this paper, we identified security flaws in two signcryption schemes proposed in [12] and [13]. Our results showed that the convertible signcryption scheme [12] fails to provide confidentiality, and the first group signcryption scheme [13] is insecure. About this specifical type of cryptosystems, the following problems seem interesting in future research: (a) presenting a formal model for group signcryption, and proposing provably secure schemes; (b) Designing schemes to support dynamic group member management in the sense that group member can join or leave the group efficiently and dynamically; (c) Optimizing the open procedure so that it does not linearly depend on the number of group members, so that such schemes are suitable for large groups.

References

1. J. H. An, Y. Dodis, and T. Rabin. On the security of joint signature and encryption. In: *EUROCRYPT 2002*, LNCS 2332, pages 83-107. Springer-Verlag, 2002.
2. G. Ateniese, J. Camenisch, M. Joye, and G. Tsudik. A practical and provably secure coalition-resistant group signature scheme. In: *CRYPTO 2000*, LNCS 1880, pages 255-270. Springer Verlag, 2000.
3. F. Bao and R.H. Deng. A signcryption scheme with signature directly verifiable by public key. In: *Public Key Cryptography* (*PKC'98*), LNCS 1431, pages 55-59. Springer-Verlad, 1998.
4. J. Baek, R. Steinfeld, and Y. Zheng. Formal proofs for the security of signcryption. In: *Public Key Cryptography* (*PKC 2002*), LNCS 2274, pages 80-98. Springer-Verlag, 2002.
5. M. Bellare and P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In: *Proc. of 1st ACM Conference on Computer and Communications Security* (*CCS'93*), pages 62-73. ACM Press, 1993.
6. X. Boyen. Multipurpose identity-based signcryption: A swiss army knife for identity-based cryptography. In: *CRYPTO'03*, LNCS 2729, pages 383-399. Springer Verlag, 2003.
7. J. Camenisch and M. Stadler. Efficient group signature schemes for large groups. In: *CRYPTO'97*, LNCS 1294, pages 410-424. Springer Verlag, 1997.
8. D. Chaum and E. van Heyst. Group signatures. In: *EUROCRYPT'91*, LNCS 950, pages 257-265. Springer-Verlag, 1992.
9. FIPS 186. *Digital Signature Standard*. U.S. Department of Commerce/NIST, National Technical Information Service, Springfield, VA, 1994.
10. T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, July 1985, IT-31(4): 469-472.
11. S. Goldwasser, S. Micali, and R. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal of Computing*, April 1988, 17(2): 281-308.
12. H.-F. Huang and C.-C. Chang. An efficient convertible authenticated encryption scheme and its variant. In: *Information and Communications Security* (*ICICS'03*), LNCS 2836, pages 382-392. Springer-Verlag, 2003.

13. D. Kwak and S. Moon. Efficient distributed signcryption scheme as group signcryption. In: *Applied Cryptography and Network Security* (*ACNS'03*), LNCS 2846, pages 403-417. Springer-Verlag, 2003.
14. B. Libert and J.-J. Quisquater. Efficient signcryption with key privacy from gap Diffie-Hellman groups. In: *Public Key Cryptography 2004*, LNCS 2947, pages 187-200. Springer-Verlag, 2004.
15. C. Ma and K. Chen. Publicly verifiable authenticated encryption. *Electronics Letters*, 39(3): 281-282, 2003.
16. J. Malone-Lee and W. Mao. Two birds one stone: signcryption using RSA. In: *CT-RSA 2003*, LNCS 2612, pages 211-225. Springer-Verlag, 2003.
17. Y. Mu, V. Varadharajan, and K. Q. Nguyen. Delegated decryption. In: *Cryptography and Coding'99*, LNCS 1746, pages 258-269. Springer Verlag, 1999.
18. Y. Mu and V. Varadharajan. Distributed signcryption. In: *INDOCRYPT 2000*, LNCS 1977, pages 155-164. Springer-Verlag, 2000.
19. D. Pointcheval and J. Stern. Security arguments for digital signatures and blind signatures. *Journal of Cryptology*, 13(3): 361-369, 2000.
20. C. Schnorr. Efficient signature generation by smart cards. *Journal of Cryptography*, 1991, 4(3): 161-174.
21. J.-B. Shin, K. Lee, and K. Shim. New DSA-verifiable signcryption schemes. In: *Information Security and Cryptology - ICISC 2002*, LNCS 2587, pages 35-47. Springer-Verlag, 2003.
22. R. Steinfeld and Y. Zheng. A signcryption scheme based on integer factorization. In: *Information Security Workshop* (*ISW'00*), LNCS 1975, pages 308-322. Springer-Verlag, 2000.
23. G. Wang, F. Bao, C. Ma, and K. Chen. Efficient authenticated encryption schemes with public verifiability. In: *Proc. of the 60th IEEE Vehicular Technology Conference* (*VTC 2004-Fall*) *- Wireless Technologies for Global Security.* IEEE Computer Society, 2004.
24. D. H. Yum and P. J. Lee. New signcryption schemes based on KCDSA. In: *Information Security and Cryptology - ICISC 2001*, LNCS 2288, pages 305-317. Springer-Verlag, 2002.
25. Y. Zheng. Digital signcryption or how to achieve cost (signature & encryption) $<<$ cost (signature) + cost (encryption). In: *CRYPTO'97*, LNCS 1294, pages 165-179. Springer-Verlag, 1997.
26. Y. Zheng. Signcryption and its application in efficient public key solution. In: *Information Security Workshop* (*ISW'97*), LNCS 1397, pages 291-312. Springer-Verlad, 1998.
27. Y. Zheng. Identification, signature and signcryption using high order residues modulo an RSA composite. In: *Public Key Cryptography* (*PKC 2001*), LNCS 1992, pages 48-63. Springer-Verlag, 2001.