On the Security of a Group Signature Scheme with Forward Security

Guilin Wang

Infocomm Security Department (ICSD)
Institute for Infocomm Research (I^2R)
21 Heng Mui Keng Terrace, Singapore 119613
http://i2r.a-star.edu.sg/icsd/staff/guilin
glwang@i2r.a-star.edu.sg

Abstract

A group signature scheme allows a group member of a given group to sign messages on behalf of the group in an anonymous and unlinkable way. In case of a dispute, however, a designated group manager can reveal the signer of a valid group signature. Based on Song's forward-secure group signature schemes, Zhang, Wu, and Wang proposed a new group signature scheme with forward security at ICICS 2003. Their scheme is very efficient in both communication and computation aspects. Unfortunately, their scheme is *insecure*. In this paper we present a security analysis to show that their scheme is *linkable*, *untraceable*, and *forgeable*.

Keywords: digital signature, group signature, forward security, cryptanalysis.

1 Introduction

In modern electronic society, digital signatures are playing an important role to provide integrity, authentication and undeniability for electronic transactions. *Group signatures*, first introduced by Chaum and van Heyst in [14], are a special king of digital signatures. In such a scheme each group member of a given group is allowed to sign messages on behalf of the group in an anonymous and unlinkable way. To check the validity of a group signature, a receiver only needs to get the unique group public key. However, with the exception of a designated *group manager*, anybody else neither can identify the identity of the signer nor determine whether multiple signatures are generated by the same group member. Furthermore, in case of later disputes, the group manager can "open" a group signature and then reveal the true signer's identity.

From the viewpoints of verifiers, only a single group public key is needed to verify group signatures. On the other hand, from the viewpoint of the signing group, its internal structure is hidden from verifiers while the signer's identities can be revealed, if necessary. In virtue of these advantages, group signatures have many potentially practical applications, such as e-voting, e-bidding and e-cash etc [15, 20, 23, 21, 13].

Following the first schemes constructed in [14], a number of new group signature schemes and improvements have been proposed [15, 9, 22, 10, 2, 3, 19, 8, 24, 12, 7]. In [15], Chen and Pedersen constructed the first scheme which allows new members to join the group dynamically, and suggested to use group signatures in e-bidding. Camenisch and Stadler proposed the first group signature scheme that can be used for large groups, since in their scheme the group public key and signatures have lengths independent of the group size [9]. Based on the strong RSA assumption [16], Camenisch and Michels presented an efficient group signature scheme in [10, 11]. Later, Kim et al. extended their scheme to support efficient member revocation [19]. Ateniese and Tsudik pointed out some obstacles that stand in the way of real world applications of group signatures, such as coalition attacks and member deletion [2]. In fact, there have been several papers which focused on the problem of member deletion [19, 8, 4]. Ateniese et al. presented a provably secure group signature scheme in [3].

Song addressed two important problems in group signature schemes, i.e., how to deal with exposure of group signing keys and how to efficiently revoke group members [24]. Here, a *group signing key* is referred to all secrets that enable a signer to produce group signatures in [24]. In fact, a group signing key consists of the membership secret and the group membership certificate [9, 3]. Based on the idea of forward secure signatures [1, 6, 17], Song constructed the fist two *forward-secure group signature* schemes. In such a scheme, the expected system life-time is divided into T time periods, and each group member's signing key evolves over time. In time period $j+1$, the signing key sk_{j+1} is updated from the signing key sk_j for time period j by using a public one-way function, and then sk_j is erased from the system. When the signing key sk_{j+1} is compromised, an attacker cannot derive any previous signing key which corresponds to a time period i $(i < j)$. Furthermore, Song also extended her schemes to support group member revocation. In Song's schemes, the group public key is affected neither by signing key update, nor by group member join or leave.

In [26], Zhang, Wu, and Wang proposed a newly efficient forward-secure group signature scheme. Since *signatures of knowledge* (refer to [9, 3]) are not used, their scheme is really very efficient in both computation and communication aspects. For example, the total computation cost of their signature generation and verification is only 7 modular exponentiations, while 36 modular exponentiations are needed in Song's schemes. At the same time, they claimed that their scheme satisfies all the desired security requirements (see Section 2). However, we find this is not the fact.

In this paper, we present a security analysis of Zhang-Wu-Wang group signature scheme with forward security [26]. More specifically, we demonstrate that their scheme is *linkable*, *untraceable* and *forgeable*. We first identify an effective way that allows anybody can determine whether two group signatures are signed by the same group member. Then, we demonstrate that any group member can forge valid signatures which cannot be opened by the group manager. This implies that their `OPEN` procedure fails to trace malicious group members. Furthermore, we prove that Zhang-Wu-Wang scheme is *untraceable in essence*, i.e., it is impossible to meet the traceability by improving `OPEN` procedure. Finally, under reasonable assumptions, a *universally forging attack* is presented. In this attack, even an outsider (not a group member) can forge valid group signatures on any messages of his choice. Therefore, Zhang-Wu-Wang scheme is *insecure*, though it is very efficient.

The rest of this paper is organized as follows. In Section 2, we introduce the informal definitions of a forward-secure group signature scheme and the security requirements. Section 3 reviews Zhang-Wu-Wang scheme [26]. Then, our security analysis is presented in Section 4. Finally, Section 5 concludes this paper.

2 Definitions

A forward-secure group signature scheme involves a *group manager*, a set of *group members*, and a set of *verifiers*. The group manager (for short, GM) is responsible for admitting/revoking group members, and for opening group signatures to reveal the true signers. When a potential user registers with GM, he/she becomes a group member and then can sign messages on behalf of the group. A verifier checks the validity of a group signature by using the unique group public key. The computational capability of each entity is modeled by a probabilistic polynomial-time Turing machine. We now review the definitions of forward-secure group signature schemes and their security requirements as follows. For more formal definitions on this subject, please refer to [7].

Definition 1. A forward-secure group signature scheme *is comprised of the following procedures [9, 2, 3, 24, 26]:*

- `SETUP`: *On input of a security parameter ℓ, this probabilistic algorithm outputs the initial group public key and the secret key for the group manager.*
- `JOIN`: *An interactive protocol between the group manager and a user that results in the user becoming a new group member. The user's output is a group signing key.*
- `SIGN`: *A probabilistic algorithm that on input a group public key, a group signing key, and a message m outputs a group signature on m.*
- `EVOLVE`: *An algorithm that on input a group signing key for time period j, outputs the corresponding group signing key for time period $j+1$.*
- `VERIFY`: *An algorithm for establishing the validity of an alleged group signature of a message with respect to a group public key.*
- `OPEN`: *An algorithm that, given a message, a valid group signature on it, a group public key and the corresponding group manger's secret key, determines the identity of the signer.*
- `REVOKE`: *An algorithm that on input a group member's certificate, a group public key and the corresponding group manger's secret key, outputs a revocation token that revokes the group member's signing ability.*

Definition 2. A forward-secure group signature scheme is secure if it satisfies all the following security requirements [2, 3, 24, 26]:

- *Correctness*: Signatures produced by a group member using `SIGN` procedure must be accepted by `VERIFY` procedure.
- *Unforgeability*: Only group members are able to sign messages on behalf of the group.
- *Anonimity*: Given a valid group signature for some message, identifying the actual signer is computationally hard for everyone but the group manager.
- *Unlinkability*: Deciding whether two different valid signatures were generated by the same group member is computationally hard for everyone but the group manager.

- *Excupability*: Even if the group manager and some of the group members collude, they cannot sign on behalf of non-involved group members.
- *Traceability*: The group manager can always open a valid group signature using `OPEN` procedure and then identify the actual signer.
- *Coalition-resistance:* A colluding subset of group members cannot generate a valid group signature that cannot be traced by the group manager.
- *Revocability*: The group manager can revoke a group member so that this group member cannot produce a valid group signature any more after being revoked.
- *Forward security*: When a group signing key is exposed, previously generated group signatures remain valid and do not need to be re-signed.

3 Review of Zhang-Wu-Wang Scheme

3.1 `SETUP` Procedure

The group manager (GM, for short) randomly chooses two large primes p_1, p_2 of the same size such that $p_1 = 2p_1' + 1$ and $p_2 = 2p_2' + 1$, where both p_1' and p_2' are also primes. Let $n = p_1 p_2$, and $G = \langle g \rangle$ be a cyclic subgroup of $\mathbb{Z}_n^*$. Usually, G is selected as the set of quadratic residues modulo n [10, 3, 24], i.e., g's order $\mathrm{ord}(g) = p_1' p_2'$. GM randomly chooses an integer x as his secret key, and then sets his public key as $y = g^x \bmod n$. GM selects a random integer e (e.g., $e = 3$) which satisfies $\gcd(e, \phi(n)) = 1$, and computes d such that $de = 1 \bmod \phi(n)$. Let $h(\cdot)$ be a publicly known coalition-resistant hash function (e.g., SHA-1, MD5). The expected system life-time is divided into T intervals and the intervals are publicly known. $(c, s) = SPK\{\gamma : y = g^\gamma\}(\ )$ denotes the signature of knowledge of $\log_g y$ on the empty message (see [9, 3] for details). Finally, the group manager publishes the public key $(y, n, g, e, h(\cdot), ID_{GM}, T)$, where ID_{GM} is the identity of the group manager.

3.2 `JOIN` Procedure

If a user, say Bob, wants to join to the group, he executes an interactive protocol with GM. Firstly, Bob chooses a random number $k \in Z_n^*$ as his secret key, and computes his identity $ID_B = g^k \bmod n$. Then, Bob generates the signature of knowledge $(c, s) = SPK\{k : ID_B = g^k\}(\ )$ to show that he knows a secret value k to meet $ID_B = g^k \bmod n$. Finally, Bob keeps k privately and sends $(ID_B, (c, s))$ to the group manager.

Upon receiving $(ID_B, (c, s))$, GM first verifies the signature of knowledge (c, s). If the verification holds, GM chooses a random number $\alpha \in Z_n^*$, and computes a triple (r_B, s_B, W_{B_0}) from

$$r_B = g^\alpha \bmod n, \quad s_B = \alpha + r_B x, \quad w_{B_0} = (r_B ID_{GM} ID_B)^{-d^T} \bmod n.$$

Then, GM sends Bob (s_B, r_B, w_{B_0}) via a private channel, and stores (s_B, r_B, w_{B_0}) together with $(ID_B, (c, s))$ in his local database. Bob accepts (s_B, r_B, w_{B_0}) as his initial membership certificate if the following two equalities hold.

$$g^{s_B} \equiv r_B y^{r_B} \bmod n, \quad \text{and} \quad r_B ID_{GM} ID_B \equiv {w_{B_0}}^{-e^T} \bmod n. \tag{1}$$

3.3 `EVOLVE` Procedure

Assume that Bob has the group membership certificate (s_B, r_B, w_{B_j}) at time period j. Then at time period $j+1$, he updates his group membership certificate as $(s_B, r_B, w_{B_{j+1}})$ by computing

$$w_{B_{j+1}} = (w_{B_j})^e \bmod n \ (= (r_B ID_{GM} ID_B)^{-d^{T-j}} \bmod n). \tag{2}$$

3.4 `SIGN` Procedure

Let (s_B, r_B, w_{B_j}) be Bob's group membership certificate at time period j. To sign a message m, Bob randomly chooses two numbers $q_1, q_2 \in \mathbb{Z}_n^*$, and computes z_1, u, r_1, r_2, r_3 as follows:

$$\begin{aligned} z_1 &= g^{q_1} y^{q_2} \bmod n, \\ u &= h(z_1, m), \\ r_2 &= w_{B_j}^u \bmod n, \\ r_1 &= q_1 + (s_B + k) \cdot u \cdot h(r_2) \text{ (in } \mathbb{Z}), \\ r_3 &= q_2 - r_B \cdot u \cdot h(r_2) \text{ (in } \mathbb{Z}). \end{aligned} \tag{3}$$

The resulting group signature on m is $\sigma = (u, r_1, r_2, r_3, m, j)$.

3.5 `VERIFY` Procedure

Given $\sigma = (u, r_1, r_2, r_3, m, j)$, a verifier accepts it as a valid group signature on m if and only if $u \equiv h(z_1', m)$, where z_1' is computed by

$$z_1' = ID_{GM}^{u \cdot h(r_2)} g^{r_1} r_2^{h(r_2) \cdot e^{T-j}} y^{r_3} \bmod n. \tag{4}$$

3.6 `OPEN` Procedure

Given a group signature $\sigma = (u, r_1, r_2, r_3, m, j)$, if necessary, GM can open it to reveal the actual identity of the signer who produced the signature. GM first checks σ's validity via `VERIFY` procedure. If σ is a valid signature, GM operates as follows to find the signer's identity:

(1) Compute $\eta = 1/(u \cdot h(r_2)) \bmod \phi(n)$.

(2) Compute $z_1' = ID_{GM}^{u \cdot h(r_2)} g^{r_1} r_2^{h(r_2) \cdot e^{T-j}} y^{r_3} \bmod n$.

(3) Search his database to find a pair (ID_B, r_B) that satisfies the following equality:

$$r_B ID_B \equiv (g^{r_1} y^{r_3} / z_1')^{\eta} \bmod n. \tag{5}$$

(4) If there is a tuple (r_B, ID_B) satisfying the above equation, GM concludes that ID_B is the identity of the actual signer. Otherwise, output $\perp$.

3.7 REVOKE Procedure

If GM wants to revoke Bob's membership certificate at time period j, he publishes a revocation token (R_j, j) in the CRL (the Certificate Revocation List), where the value R_j is computed by

$$R_j = (r_B ID_B)^{d^{T-j}} \bmod n. \tag{6}$$

Given a valid group signature $\sigma = (u, r_1, r_2, r_3, m, j)$, a verifier can identify whether σ is produced by a revoked group member. For this sake, he performs as follows:

(1) Compute $z_1' = ID_{GM}^{u \cdot h(r_2)} g^{r_1} r_2^{h(r_2) \cdot e^{T-j}} y^{r_3} \bmod n$.

(2) Search all (R_i, i) $(i \leqslant j)$ in CRL to check whether there is a R_i $(i \leqslant j)$ such that the following equality holds:

$$g^{r_1} y^{r_3} \equiv z_1' (R_i^{e^{T-i}})^{u \cdot h(r_2)} \bmod n. \tag{7}$$

(3) If one such R_i $(i \leqslant j)$ is found, the verifier concludes that the signature σ is revoked, i.e., it is generated by a group member after he is revoked.

4 Security Analysis of Zhang-Wu-Wang Scheme

In [26], Zhang et al. analyzed the security of their scheme, and concluded that their scheme satisfies all the security requirements listed in Section 2. However, we find that this is not the fact.

4.1 Linkability

Let $\sigma = (u, r_1, r_2, r_3, m, j)$ and $\bar{\sigma} = (\bar{u}, \bar{r}_1, \bar{r}_2, \bar{r}_3, \bar{m}, j)$ be two (valid) group signatures. To decide whether they are signed by the same group member, a verifier only need to check whether the following equality holds.

$$r_2^{\bar{u}} \equiv \bar{r}_2^{u} \bmod n. \tag{8}$$

In fact, if both σ and $\bar{\sigma}$ are signed by the same group member, say Bob, according to SIGN procedure, we know that $r_2^{\bar{u}} = (w_{B_j})^{u \cdot \bar{u}} \bmod n = \bar{r}_2^{u} \bmod n$. So the above equality holds for σ and $\bar{\sigma}$.

On the other hand, we can show that if σ and $\bar{\sigma}$ are signed by two different group members, say Bob and Charlie, respectively, equation (8) unlikely holds. To prove this claim, on the contrary, we assume that σ and $\bar{\sigma}$ satisfy equation (8). Let $r_B = g^{\alpha} \bmod n$, $r_C = g^{\bar{\alpha}} \bmod n$, $ID_B = g^k \bmod p$, and $ID_C = g^{\bar{k}} \bmod p$. Since σ is signed by Bob, and $\bar{\sigma}$ is signed by Charlie, we have $r_2^{\bar{u}} = (w_{B_j})^{u \cdot \bar{u}} \bmod n$, and $\bar{r}_2^{u} = (w_{C_j})^{u \cdot \bar{u}} \bmod n$. From $r_2^{\bar{u}} = \bar{r}_2^{u} \bmod n$, we have $(r_B ID_{GM} ID_B)^{-u\bar{u}d^{T-j}} = (r_C ID_{GM} ID_C)^{-u\bar{u}d^{T-j}} \bmod n$. So, the following equation holds

$$g^{(k - \bar{k} + \alpha - \bar{\alpha}) u\bar{u} d^{T-j}} = 1 \bmod n. \tag{9}$$

Since $\mathrm{ord}(g) = p_1' p_2'$, $\gcd(d, \phi(n)) = 1$, and $\phi(n) = 4p_1' p_2'$, we know $\mathrm{ord}(g) \nmid d$. At the same time, usually $|h(\cdot)| = 160$, and $|p_1'| = |p_2'| \geqslant 255$, thus we also have $\mathrm{ord}(g) \nmid u\bar{u}$. Therefore, from equation (9), we conclude that $\mathrm{ord}(g) | (k - \bar{k} + \alpha - \bar{\alpha})$, i.e., $(k - \bar{k} + \alpha - \bar{\alpha}) =$

0 or $(k - \bar{k} + \alpha - \bar{\alpha}) = b \cdot \mathrm{ord}(g)$ for some non-zero integer b. However, both cases unlikely happen, because $\mathrm{ord}(g)$ is the product of two large primes, α and $\bar{\alpha}$ are random numbers selected by GM, and k and $\bar{k}$ are random numbers selected by Bob and Charlie respectively.

Furthermore, equation (8) can be generalized to link two signatures which are generated by the same group member at different time periods. That is, given two group signatures $\sigma = (u, r_1, r_2, r_3, m, j)$ and $\bar{\sigma} = (\bar{u}, \bar{r}_1, \bar{r}_2, \bar{r}_3, \bar{m}, i)$ where $j > i$, one can know whether the same group member generated them by checking

$$r_2^{\bar{u}} \equiv \bar{r}_2^{u \cdot e^{j-i}} \bmod n. \tag{10}$$

4.2 Untraceability

In this section, we demonstrate an attack that enables a group member Bob to forge a valid certificate. Then, using this forged certificate, he can generate valid group signature on any message of his choice without being traced. Firstly, we note that it seems difficult to forge a new pair (r_B, s_B) so that the first equation in (1) is satisfied, since Bob does not know GM's secret key x. However, Bob can change the values of w_{B_0} and ID_B, and get a new certificate. For this sake, he chooses a random number $a \in \mathbb{Z}_n$, and defines $\bar{w}_{B_0}$, $\overline{ID}_B$ and $\bar{k}$ as follows:

$$\bar{w}_{B_0} = w_{B_0} g^a \bmod n,\ \overline{ID}_B = ID_B \cdot g^{-ae^T} \bmod n,\ \bar{k} = k - ae^T \text{ (in } \mathbb{Z}). \tag{11}$$

In the following, we show that the tuple $(s_B, r_B, \bar{w}_{B_0})$ with respect to $(\overline{ID}_B, \bar{k})$ constitutes a valid group membership certificate. Firstly, $\overline{ID}_B = ID_B g^{-a \cdot e^T} \bmod n = g^{k - a \cdot e^T} \bmod n = g^{\bar{k}} \bmod n$. Secondly, we already have $g^{s_B} = r_B y^{r_B} \bmod n$. Finally, the following equalities hold

$$\begin{aligned} r_B ID_{GM} \overline{ID}_B &= (r_B ID_{GM} ID_B) \cdot g^{-ae^T} \bmod n \\ &= {w_{B_0}}^{-e^T} \cdot g^{-ae^T} \bmod n \\ &= (w_{B_0} \cdot g^a)^{-e^T} \bmod n \\ &= \bar{w}_{B_0}^{-e^T} \bmod n. \end{aligned}$$

Therefore, according to `JOIN` procedure, Bob obtains another new certificate $(s_B, r_B, \bar{w}_{B_0})$ with $(\overline{ID}_B, \bar{k})$. Using this tuple $(s_B, r_B, \bar{w}_{B_0}, \overline{ID}_B, \bar{k})$, Bob can generate valid group signatures on arbitrary messages. According to `OPEN` procedure, when such forged signatures are presented, Bob will not be traced as the signer since $r_B \overline{ID}_B \neq r_B ID_B \bmod n$. Therefore, the `OPEN` procedure provided by [26] fails to trace such malicious group members. A natural question is that whether can we improve this `OPEN` procedure such that it can reveal the identities of malicious group members? Unfortunately, the answer is negative. In other words, Zhang-Wu-Wang scheme is *untraceable in essence*, i.e., two malicious members may forge the same valid but untraceable group signature on a given message. More formally, we have the following theorem.

Theorem 1. *Using the above attack, the forged group signatures generated by two malicious members for the same message are perfectly indistinguishable.*

Proof: We only need to prove that if Bob forges a group signature σ on a message m, Charlie can also generate it by using our above attack. For simplicity, let $sk_{B,j} = (s_B, r_B, w_{B_j}, ID_B, k)$, and $fsk_{B,j} = (s_B, r_B, \bar{w}_{B_j}, \overline{ID}_B, \bar{k})$, where $w_{B_j} = w_{B_{j-1}}^e \bmod n =$

$w_{B_0}^{e^j} \bmod n$, and $\bar{w}_{B_j} = \bar{w}_{B_{j-1}}^e \bmod n = \bar{w}_{B_0}^{e^j} \bmod n$. $sk_{B,j}$ and $fsk_{B,j}$ denote Bob's *signing key* and *forged signing key* at time period j, respectively. Here, $\bar{w}_{B_0}, \overline{ID}_B$ and $\bar{k}$ are computed by Bob as in the above attack, i.e., Bob selects a random number a and calculates the values of them from equation (11). To forge a group signature on message m, he randomly chooses two numbers $q_1, q_2 \in_R \mathbb{Z}_n^*$, and computes $z_1 = g^{q_1} y^{q_2} \bmod n$, $u = h(z_1, m)$, $r_2 = \bar{w}_{B_j}^u \bmod n$, $r_1 = q_1 + (s_B + \bar{k})uh(r_2)$ (in $\mathbb{Z}$), and $r_3 = q_2 - r_B uh(r_2)$ (in $\mathbb{Z}$). $\sigma = (u, r_1, r_2, r_3, m, j)$ is the resulting forged group signature on message m.

Now, we show that Charlie can forge a group signature σ' on the same message m such that $\sigma' \equiv \sigma$, if he chooses an appropriate number a' to define his forged signing key, and two specific numbers q_1' and q_2' to produce group signature. Let $sk_{C,j} = (s_C, r_C, w_{C_j}, ID_C, k')$ be Charlie's signing key at time period j, where $s_C = \alpha' + r_C x$, $r_C = g^{\alpha'} \bmod n$, $ID_C = g^{k'} \bmod n$, $w_{C_j} = w_{C_0}^{e^j} \bmod n$, and $w_{C_0} = (r_C ID_{GM} ID_C)^{-d^T} \bmod n$. To forge a new membership certificate, Charlie first sets $a' = a - (k + \alpha - k' - \alpha') \cdot e^{-T} \bmod \mathrm{ord}(g)$. This means that there exists an integer l such that $k' + \alpha' - a'e^T = k + \alpha - ae^T + l \cdot \mathrm{ord}(g)$. And then, according to equation (11), he defines $\bar{w}_{C_0}, \overline{ID}_C$ and $\bar{k}'$ as follows:

$$\bar{w}_{C_0} = w_{C_0} g^{a'} \bmod n,\ \overline{ID}_C = ID_C g^{-a'e^T} \bmod n,\ \bar{k}' = k' - a'e^T \text{ (in } \mathbb{Z}). \tag{12}$$

Up to this, Charlie obtains his forged signing key $fsk_{C,j} = (s_C, r_C, \bar{w}_{C_j}, \overline{ID}_C, \bar{k}')$ for time period j, where $\bar{w}_{C_j} = \bar{w}_{C_{j-1}}^e \bmod n = \bar{w}_{C_0}^{e^j} \bmod n$. Due to the specific choice of the value of a', we have

$$r_B \overline{ID}_B = r_C \overline{ID}_C \bmod n, \quad \text{and} \quad \bar{w}_{B_0} = \bar{w}_{C_0} \bmod n.$$

To forge a group signature σ' on the message m such that $\sigma' \equiv \sigma = (u, r_1, r_2, r_3, m, j)$, Charlie first sets $q_1' = q_1 + xuh(r_2)(r_B - r_C) - l \cdot uh(r_2) \cdot \mathrm{ord}(g)$, and $q_2' = q_2 - uh(r_2)(r_B - r_C)$. Then, he computes $z_1' = g^{q_1'} y^{q_2'} \bmod n$, $u' = h(z_1', m)$, $r_2' = \bar{w}_{C_j}^{u'} \bmod n$, $r_1' = q_1' + (s_C + \bar{k}')u'h(r_2')$ (in $\mathbb{Z}$), and $r_3' = q_2' - r_C u'h(r_2')$ (in $\mathbb{Z}$). Let $\sigma' = (u', r_1', r_2', r_3', m, j)$ be the resulting group signature forged by Bob. Then, one can directly verify that $z_1' = z_1$, $u' = u$, $r_1' = r_1$, $r_2' = r_2$ and $r_3' = r_3$. In other words, $\sigma' \equiv \sigma$.

The above discussion shows that for a given forged group signature σ, any group member may be the attacker who forged it by using our attack. Therefore, Theorem 1 holds[1].

4.3 Forgeability

The attack given in Section 4.2 only enables group members to forge valid group signatures. This Section demonstrates a universal forgery which can be mounted by anybody, not necessarily the group members. We first describe our attack when the value of $ID_{GM}^{-d^{T-j}} \bmod n$ is available, and then explain how to get the value of $ID_{GM}^{-d^{T-j}} \bmod n$ using some public information.

If the value of $ID_{GM}^{-d^{T-j}} \bmod n$ is known, to forge a group signature on an arbitrary message m, the following *universally forging attack* can be mounted by anybody.

[1]Note that in the proof of Theorem 1, we only discuss the *possibility* whether two group members can forge the same valid but untraceable group signature on the same message, so Charlie can 'use' some secrets controlled by other parties, such as the values of a, q_1, q_2, x and $\mathrm{ord}(g)$ etc.

(1) Select four random numbers $a, b, q_1, q_2 \in_R \mathbb{Z}_n^*$.

(2) Compute $z_1 = g^{q_1} y^{q_2} \bmod n$, and $u = h(z_1, m)$.

(3) Define $r_2 = (ID_{GM}^{-d^{T-j}})^u g^{-a} y^b \bmod n$, $r_1 = q_1 + ah(r_2)e^{T-j}$ (in $\mathbb{Z}$), and $r_3 = q_2 - bh(r_2)e^{T-j}$ (in $\mathbb{Z}$).

(3) Output $\sigma = (u, r_1, r_2, r_3, m, j)$ as the forged group signature on message m.

We now show the correctness of our above attack. According to `VERIFY` procedure, we need to first compute z_1' and then checks whether $u \equiv h(z_1', m)$. By equation (4), we have the following equalities:

$$\begin{aligned} z_1' &= ID_{GM}^{uh(r_2)} g^{r_1} r_2^{h(r_2)e^{T-j}} y^{r_3} \bmod n \\ &= ID_{GM}^{uh(r_2)} g^{r_1} ((ID_{GM}^{-d^{T-j}})^u g^{-a} y^b)^{h(r_2)e^{T-j}} y^{r_3} \bmod n \\ &= g^{r_1 - ah(r_2)e^{T-j}} y^{r_3 + bh(r_2)e^{T-j}} \bmod n \\ &= g^{q_1} y^{q_2} \bmod n \\ &= z_1 \bmod n. \end{aligned}$$

So, $z_1' = z_1$. This means $u = h(z_1, m) = h(z_1', m)$, i.e., our above attack succeeds.

Now, we describe an algorithm which enables an outsider Alice to derive the value of $ID_{GM}^{-d^{T-j}} \bmod n$ alone from a number of known group signatures, revocation tokens and the group public key. Before presenting the details, we first explain the basic idea. Assume that the attacker Alice obtains two group signatures $\sigma = (u, r_1, r_2, r_3, m, i)$ and $\sigma' = (u', r_1', r_2', r_3', m', i)$, which are generated by Bob at time period i and satisfy $\gcd(u, u') = 1$. Later, GM revokes Bob by releasing a revocation token (R_j, j) at time period j $(i < j)$. Since $\gcd(u, u') = 1$, by using extended Euclidian algorithm Alice can get two integers a and b such that $au + bu' = 1$. Then, she gets the value of w_{B_i} by

$$w_{B_i} = (r_2)^a \cdot (r_2')^b \bmod n. \tag{13}$$

Equation (13) holds because we have $w_{B_i} = w_{B_i}^1 \bmod n = w_{B_i}^{au+bu'} \bmod n = (w_{B_i}^u)^a (w_{B_i}^{u'})^b \bmod n = (r_2)^a \cdot (r_2')^b \bmod n$. Finally, Alice computes the value of $ID_{GM}^{-d^{T-j}} \bmod n$ by

$$ID_{GM}^{-d^{T-j}} = (w_{B_i})^{e^{j-i}} \cdot R_j \bmod n. \tag{14}$$

Equation (14) is justified by the following equalities:

$$\begin{aligned} ID_{GM}^{-d^{T-j}} &= (ID_{GM} r_B ID_B)^{-d^{T-j}} \cdot (r_B ID_B)^{d^{T-j}} \bmod n \\ &= (ID_{GM} r_B ID_B)^{-d^{T-i} \cdot e^{j-i}} \cdot R_j \bmod n \\ &= (w_{B_i})^{e^{j-i}} \cdot R_j \bmod n. \end{aligned}$$

In the following complete description of our algorithm, Alice also has to find who is the signer revoked by (R_j, j).

Step 1). The attacker Alice collects a number of valid group signatures, and uses the method described in Section 4.1 to classify them into different directories, denoted by D_l $(1 \leqslant l \leqslant n)$. Here, the signatures in each directory are generated by a different signer, i.e., a different group member. For future use, she stores all classified signatures in her local database.

Step 2). Once a revocation token (R_j, j) is released, she computes $P = R_j^{e^{T-j}} \bmod n$. Note that if Bob is the group member revoked by (R_j, j), it is easy to know that $P = r_B ID_B \bmod n$.

Step 3). Alice chooses one signature $\sigma = (u, r_1, r_2, r_3, m, i)$ from each non-empty directory D_l, and searches which signature satisfies the following equality:

$$r_2^{e^{T-i}} \equiv (ID_{GM} \cdot P)^{-u} \bmod n.$$

If one such signature $\sigma \in D_{\bar{l}}$ is found, Alice concludes that the group member (say Bob) corresponding to the directory $D_{\bar{l}}$ is revoked.

Step 4). Alice searches the directory $D_{\bar{l}}$ in her local database to find two group signatures $\sigma = (u, r_1, r_2, r_3, m, i)$, and $\sigma' = (u', r_1', r_2', r_3', m', i')$ such that $i' = i < j$ and $\gcd(u', u) = 1$. If two such signatures (generated by Bob) are found, Alice executes the expended Euclidian algorithm to get two integers a and b such that $au + bu' = 1$. Then, Alice can get the value of w_{B_i} $(i < j)$ from equation (13).

Step 5). Finally, by using the values of w_{B_i} $(i < j)$ and R_j, Alice derives the value of $ID_{GM}^{-d^{T-j}} \bmod n$ by equation (14).

The success of the above algorithm depends on the following assumption: Alice can find two group signatures σ and σ' such that $\gcd(u', u) = 1$, and that they are generated during the same time period by the same group member who is revoked later. There are several reasons to support that this is a reasonable assumption in practical applications. Firstly, a group member of course may generate a number of signatures during the same time period, and be revoked later. Secondly, for two randomly selected integers N and M, $\gcd(N, M) = 1$ happens with a very high probability $6/\pi^2 \approx 0.6$ [27]. Since the hash function $h(\cdot)$ can be viewed as a random function with fixed bit-length output (e.g. 160-bit), u' and u can be treated as random numbers. Under this treatment, $\gcd(u', u) = 1$ also holds with probability about 0.6.

5 Conclusion

In this paper, we presented a security analysis of Zhang-Wu-Wang group signature scheme proposed in [26]. By successfully identifying several attacks, we demonstrated that their scheme is *insecure*. More specifically, our results shows that their scheme is *linkable*, *untraceable* and *forgeable*. In fact, how to design a secure and more efficient group signature scheme is still a hot problem in this area. The most recent investigations are given in [5, 7, 18, 25].

References

[1] R. Anderson. Invited Lecture, *4th ACM Computer and Communications Security*, 1997.

[2] G. Ateniese and G. Tsudik. Some open issues and new directions in group signature schemes. In: *Financial Cryptography (FC'99), LNCS 1648*, pages 196-211. Springer-Verlag, 1999.

[3] G. Ateniese, J. Camenisch, M. Joye, and G. Tsudik. A practical and provably secure coalition-resistant group signature scheme. In: *Advances in Cryptology - CRYPTO'2000, LNCS 1880*, pages 255-270. Springer-Verlag, 2000.

[4] G. Ateniese, D. Song, and G. Tsudik. Quasi-efficient revocation of group signatures. In: *Financial Cryptography (FC'02), LNCS 2357*. Springer-Verlag, 2002. Primary version available at `http://eprint.iacr.org/2001/101/`

[5] G. Ateniese and B. de Medeiros. Efficient group signatures without trapdoors. In: ASIACRYPT 2003 (to appear). Springer-Verlag, 2003. Primary version available at `http://eprint.iacr.org/2002/173/`

[6] M. Bellare, and S. Miner. A forward-secure digital signature scheme. In: *Advances in Cryptology - CRYPTO'99, LNCS 1666*, pp. 431-448. Springer-Verlag, 1999.

[7] M. Bellare, D. Micciancio, and B. Warinschi. Foundations of group signatures: formal definitions, simplified requirements, and a construction based on general assumptions. In: *Advances in Cryptology - EUROCRYPT'03, LNCS 2656*, pp. 614-629. Springer-Verlag, 2003.

[8] E. Bresson and J. Stern. Efficient revocation in group signatures. In: *Public Key Cryptography (PKC'01), LNCS 1992*, pages 190-206. Springer-Verlag, 2001.

[9] J. Camenisch and M. Stadler. Effient group signature schemes for large groups. In: *Advances in Cryptology - CRYPTO'97, LNCS 1294*, pages 410-424. Springer-Verlag, 1997.

[10] J. Camenisch and M. Michels. A group signature scheme with improved efficiency. In: *Advances in Cryptology - ASIACRYPT'98, LNCS 1514*, pages 160-174. Springer-Verlag, 1998.

[11] J. Camenisch and M. Michels. A group signature scheme based on an RSA-variant. *Technical Report RS-98-27*, BRICS, University of Aarhus, November 1998. An earlier version appears in [10].

[12] J. Camenisch and A. Lysyanskaya. Dynamic accumulators and application to efficient revocation of anonymous credentials. In: *Advances in Cryptology - CRYPTO 2002, LNCS 2442*, pages 61-76. Springer-Verlag, 2002.

[13] S. Canard and J. Traoré. On fair e-cash systems based on group signature schemes. In: *Information Security and Privacy (ACISP 2003), LNCS 2727*, pp. 237-248. Berlin: Springer-Verlag, 2003.

[14] D. Chaum and E. van Heyst. Group signatures. In: *Advances in Cryptology - EUROCRYPT'91, LNCS 950*, pages 257-265. Springer-Verlag, 1992.

[15] L. Chen and T. P. Pedersen. New group signature schemes. In: *Advances in Cryptology - EUROCRYT'94, LNCS 950*, pages 171-181. Springer-Verlag, 1995.

[16] E. Fujisaki and T. Okamoto. Statistical zero-knowledge protocols to prove modular polynomial relations. In: *Advances in Cryptology - CRYPTO'97, LNCS 1294*, pages 16-30. Springer-Verlag, 1997.

[17] G. Itkis and L. Reyzin. Forward-secure signatures with optimal signing and verifying. In: *Advances in Cryptology - CRYPTO'01 LNCS 2139*, pages 332-354. Springer-Verlag, 2001.

[18] A. Kiayias and M. Yung. Extracting group signatures from traitor tracing schemes. In: *Advances in Cryptology - EUROCYPT 2003, LNCS 2656*, pp. 630-648. Springer-Verlag, 2003.

[19] H.J. Kim, J.I. Lim, and D.H. Lee. Efficient and secure member deletion in group signature schemes. In: *Information Security and Cryptology (ICISC 2000), LNCS 2015*, pages 150-161. Springer-Verlag, 2001.

[20] A. Lysyanskaya and Z. Ramzan. Group blind digital signatures: A scalable solution to electronic cash. In: *Financial Cryptography (FC'98), LNCS 1465*, pages 184-197. Springer-Verlag, 1998.

[21] G. Maitland and C. Boyd. Fair electronic cash based on a group signature scheme In: *Information Security and Cryptography (ICICS'01), LNCS 2229*, pp. 461-465. Springer-Verlag: 2001.

[22] H. Petersen. How to convert any digital signature scheme into a group signature scheme. In: *Security Protocols Workshop, LNCS 1361*, pages 177-190. Springer-Verlag, 1997.

[23] K. Sakurai and S. Miyazaki. An anonymous electronic bidding protocol based on a new convertible group signature scheme. In: *Information Security and Privacy (ACISP'00), LNCS 1841*, pp. 385?99. Springer-Verlag, 2000.

[24] D.X. Song. Practical forward secure group signature schemes. In: *Proceedings of the 8th ACM Conference on Computer and Communications Security (CCS 2001)*, pages 225-234. ACM press, 2001.

[25] G. Tsudik and S. Xu. Accumulating composites and improved group signing. In: *ASIACRYPT 2003* (to appear). Springer-Verlag, 2003. Primary version available at `http://eprint.iacr.org/2003/112/`

[26] J. Zhang, Q. Wu, and Y. Wang. A novel efficient group signature scheme with forward security. In: *Information and Communications Security (ICICS'03), LNCS 2836*, pages 292-300. Springer-Verlag, 2003.

[27] `http://mathworld.wolfram.com/RelativelyPrime.html`