# Security of Popescu Group Signature Scheme for Large Groups

*This work is a project supported by Scientific Research Fund of Zhejiang Provincial Education Department.*

**Zuhua Shao**

Department of Computer and Electronic Engineering

Zhejiang University of Science and Technology

Hangzhou, Zhejiang, 310012,

P. R. of CHINA

**Abstract**: Group signature is a primitive in digital signatures. A group signature allows members of a group to sign messages anonymously on behalf of the group. In the event of a dispute, a designated trusted entity can reveal the identity of the actual signer. The application of group signatures includes e-voting, e-bidding and e-cash. Recently Popescu proposed an efficient group signature scheme for large groups. The lengths of the group's public key and of signatures do not depend on the size of the group. In this paper, however, we find there are some problems in the Popescu group signature scheme. Moreover, this scheme is vulnerable to forgery attacks. Any adversary could easily forge a valid group signature for any message without the knowledge of the secret values of the legal members of the group.

**Indexing terms:** cryptography, group signature, discrete logarithm, factoring, forgery attack

**Zuhua Shao** was born in Shanghai, People's Republic of China, On 30 April 1948. He received B.S. degree in mathematics and M.S. in algebra from the Northeastern Normal University, People's Republic of China in 1976 and 1981 respectively. Since 1990 he has taught computer science as an associate professor in the Hangzhou Institute of Financial Managers, The Industrial and Commerce Bank of China. Now he is a teacher at the Zhejiang University of Science and Technology. His current research interests are cryptography and financial data security.

## 1. Introduction

Group signature is a relatively new cryptographic concept, which allows and only allows individual member of a group to compute digital signature for any message on behalf of the group. The receivers can verify that it is a valid group signature with respect to a single group public key, but cannot identify which member of the group issued it. Furthermore, group signatures are unlikable, which makes it computationally hard to establish whether or not multiple signatures are produced by the same group member. In the case of a late dispute, the group manager can reveal the identity of the signer. The group signatures can be applied to e-voting, e-bidding and e-cash, where the privacy of signers or of organizations is required.

In 1992, Chaum and Heyst [1] introduced the first group signature scheme. Since then, some group signature schemes were proposed [2-8]. Some were based on factoring (FAC), and others were based on discrete logarithm (DL).

In 2001, Popescu [9] proposed a new efficient group signature scheme based on Okamoto-Shiraishi's assumption [10] (the difficulty of computation of approximate $e$-th roots modulo a composite number). One main advantage of the Popescu group signature scheme is that both the lengths of the group public key and the group signatures are independent of the group's size. Hence the Popescu group signature scheme is well suitable for large groups.

In this paper, we would like to point out that there are some problems in the Popescu group signature scheme. We would also show that this scheme is not secure since any adversary can easily forge a valid group signature for any message without the secret values of the legal members of the group. The verifiers

could not find this forgery, even the group manager could not find who is the actual signer. The reason of security problem is that the membership certification is not verified.

## 2. Brief Review of the Popescu Group Signature Scheme

A group signature scheme consists of five phases: (1) Setup: the group manager chooses the private parameter and the public key of the group. (2) Join: a group member joins the group by obtaining his membership certificate. (3) Sign: a group member signs messages with his private key. (4) Verification: any verifier validates the group signature with respect to a single group public key. (5) Open: the group manager determines the identity of the actual signer of a group signature.

### 2.1 Setup

The group manager chooses random prime numbers $p'$, $q'$ and computes $p=2p'+1$ and $q=2q'+1$. Then the group manager computes $n = pq$. Let $l_n$ denote the bit-length of $n$. He chooses a real $\varepsilon > 1$ and a public exponent $e > 4$ such that $e$ is relatively prime to $\varphi(n)$. The group manager selects $g$ an element of $Z_n^*$ of order $n$. Let $G = <g>$. The group manager selects an element $h \in G$ whose discrete logarithm to the base $g$ must not be known. He chooses an element $C \in Z_n^*$. The group manager chooses a secret value $x$ and computes $y = g^x \pmod n$. Finally, a collision-resistant hash function $H:\{0,1\}^{*?} \rightarrow \{0,1\}^k$ and security parameters $?>1$, $l_1$, $l_2$ are set. The public key is $P = (n, \varepsilon, e, g, y, h, C, \ell_n, ?, l_1, l_2)$ and the secret key is $S=(p', q', x)$. In practice, the component of $P$ must be verifiable to prevent framing attacks.

### 2.2 Join

Suppose now that a user wants to join the group. We assume that communication between a group member and the group manager is secure, i. e. private and authentic. A membership certificate in the group signature scheme consists of a pair of integers $(X, \delta)$ satisfying $X^e = C + \delta \pmod n$ and $\delta \in [2^{l_1}, 2^{l_1} + 2^{l_2} - 1]$. To obtain his membership certification, each user $U_i$ must perform the following protocol with the group manger.

1. The user $U_i$ selects a random element $x_i \in [2^{l_1}, 2^{l_1} + 2^{l_2} - 1]$ and computes $ID_i = g^{x_i} \pmod n$
2. The user $U_i$ must prove to the group manager that he knows $Dlog_g ID_i$ and that this value is in the interval $(2^{l_1} - 2\varepsilon^{(l_2+k)+1}, 2^{l_1} + 2\varepsilon^{(l_2+k)+1})$
3. Then the user $U_i$ chooses a random number $r \in Z_n^*$ and computes $z = r^e(C + x_i) \pmod n$. He sends $z$ to the group manager.
4. The group manager computes $v = z^{1/e} \pmod n = r(C + x_i)^{1/e} \pmod n$ and sends $v$ to the user $U_i$.
5. The user $U_i$ *computes* $A_i = v/r = (C + x_i)^{1/e} \pmod n$. The pair $(A_i, X_i)$ is the member certificate of the user $U_i$.

Consequently, at the end of the protocol, the group manager does not know the membership certification $(A_i, X_i)$ of the user $U_i$. The group manager creates a new entry in the group database and stores $ID_i$ in the new entry.

## 2.3 Sign

A group member $U_i$, with a member certificate $(A_i, X_i)$, can generate anonymous and unlinkable group signature on a message $m$ as follows:

1. Choose an integer $w \in_R \{0,1\}^{l_2}$ and computes

   $A = A_i h^w \pmod n$, $B = g^w \pmod n$, $D = g^{x_i} y^w \pmod n$.

2. Choose $r_1 \in_R \{0,1\}\varepsilon^{(l_2+k)}$, $r_2 \in_R \{0,1\}\varepsilon^{(l_G+l_1+k)}$, $r_3 \in_R \{0,1\}\varepsilon^{(l_G+k)}$, $r_4 \in_R \{0,1\}\varepsilon^{(l_2+k)}$, $r_5 \in_R \{0,1\}\varepsilon^{(l_2+k)}$ and computes

   $d_1 = B^{r_1} / g^{r_2} \pmod n$

   $d_2 = g^{x_i^2} D^{r_4} / y^{r_5} \pmod n$

   $d_3 = g^{r_3} \pmod n$

   $d_4 = g^{r_1} y^{r_3} \pmod n$

3. Compute $c = H(m||g||h||y||A||B||D||d_1||d_2||d_3||d_4)$
4. Compute $s_1 = r_1 - c(x_i - 2^{l_1})$, $s_2 = r_2 - cx_i w$, $s_3 = r_3 - cw$, $s_4 = r_4 + x_i + c2^{l_1}$, $s_5 = r_5 + x_i w + c2^{l_1}$ (in $Z$)
5. Sends the group signature $(c, s_1, s_2, s_3, s_4, s_5, A, B, D)$ to the verifier.

## 2.4 Verify

The resulting signature $(c, s_1, s_2, s_3, s_4, s_5, A, B, D)$ of the message $m$ can be verified as follows:

1. Compute $c' = H(m||g||h||y||A||B||D|| B^{s_1-c2^{l_1}} / g^{s_2} \pmod n || D^{s_4-c2^{l_1}} / y^{s_5-c2^{l_1}} \pmod n || B^c g^{s_3} \pmod n || D^c g^{s_1-c2^{l_1}} y^{s_3} \pmod n)$
2. Accepted the group signature $(c, s_1, s_2, s_3, s_4, s_5, A, B, D)$ if and only if $c = c'$ and

   $s_1 \in_R \{-2^{(l_2+k)}, ..., 2\varepsilon^{(l_2+k)}\}$, $s_2 \in_R \{2^{l_G+l_1+k}, ..., 2\varepsilon^{(l_G+l_1+k)}$, $s_3 \in_R \{2^{l_G+k}, ..., 2\varepsilon^{(l_G+k)}$, $s_4 \in_R \{2^{(l_2+k)}, ..., 2\varepsilon^{(l_2+k)}\}$, $s_5 \in_R \{2^{(l_2+k)}, ..., 2\varepsilon^{(l_2+k)}\}$

## 2.5 Open

Given a group signature $(c, s_1, s_2, s_3, s_4, s_5, A, B, D)$, the group manager can, by checking its correctness, find out which one of the group members issued this signature. He gives up if the group signature is not correct.

Otherwise, he performs the following steps:

1. Recover $ID_i$ (the identity of the user $U_i$) as $ID_i = D/B^x \pmod n$
2. Prove that $Dlog_g y = Dlog_B(D/ID_i \bmod n)$

# 3. Security analyses

## 3.1 The choice of the public key of the group

The group manager chooses random prime numbers $p'$, $q'$ and computes $p=2p'+1$ and $q=2q'+1$. Then the group manager computes $n = pq$. However that $p'$ and $q'$ are prime numbers does not implies that $p$ and $q$ are also prime. If $n$ contain several small factors, the computation of approximate $e$-th root modulo the composite number $n$ is not difficult, so is the computation of the discrete logarithm $Dlog_g ID_i$.

Even if $p$ and $q$ are prime, the group manager can also compute the discrete logarithm $Dlog_g ID_i$ with the knowledge of the factoring $n$ as follows:

1. Compute two integers $a$ and $b$ such that $ID_i = g^a \pmod p$ and $ID_i = g^b \pmod q$ since $l_n = 1200$ implies $l_p = 600$ and $l_q = 600$. So $x_i = a \bmod 2p'$ and $x_i = b \bmod 2q'$
2. Compute $x$ such that $x = a \bmod 2p'$ and $x = b \bmod q'$ by the Chinese Remainder Theorem, since $p'$ and $q'$ are also prime. Hence $x = x_i$.
3. Compute $A_i = (C + x_i)^{1/e} \pmod n$

Therefore the group manager with the knowledge of the membership certificate $(A_i, x_i)$ can sign any message on behalf of the user $U_i$.

Besides, there does not exist any element of order $n$ in $Z_n^*$. The group manager cannot select $g$ an element of $Z_n^*$ of order $n$.

## 3.2 Forgery attack

Suppose that an adversary does not know any membership certificate $(A_i, x_i)$. To forge a group signature for any message $m$, he does the following steps:

1. Selects two random elements $v$ and $u \in [2^{l_1},\ 2^{l_1} + 2^{l_2} - 1]$ as forged membership certificate.
2. Choose an integer $w \in_R \{0,1\}^{l_2}$ and computes

   $A = vh^w \pmod n$, $B = g^w \pmod n$, $D = g^u y^w \pmod n$.
3. Choose $r_1 \in_R \{0,1\}\varepsilon^{(l_2+k)}$, $r_2 \in_R \{0,1\}\varepsilon^{(l_G+l_1+k)}$, $r_3 \in_R \{0,1\}\varepsilon^{(l_G+k)}$, $r_4 \in_R \{0,1\}\varepsilon^{(l_2+k)}$, $r_5 \in_R \{0,1\}\varepsilon^{(l_2+k)}$ and computes

   $d_1 = B^{r_1} / g^{r_2} \pmod n$

   $d_2 = g^{u^2} D^{r_4} / y^{r_5} \pmod n$

   $d_3 = g^{r_3} \pmod n$

   $d_4 = g^{r_1} y^{r_3} \pmod n$
4. Compute $c = \mathrm{H}(m||g||h||y||A||B||D||d_1||d_2||d_3||d_4)$
5. Compute $s_1 = r_1 - c(u - 2^{l_1})$, $s_2 = r_2 - cuw$, $s_3 = r_3 - cw$, $s_4 = r_4 + u + c2^{l_1}$, $s_5 = r_5 + uw + c2^{l_1}$ (in $Z$)
6. Send the group signature $(c, s_1, s_2, s_3, s_4, s_5, A, B, D)$ to the verifier.

Upon receiving the message ($c$, $s_1$, $s_2$, $s_3$, $s_4$, $s_5$, $A$, $B$, $D$), the verifier computes

$$d_1' = B^{s_1 - c2^{l_1}} / g^{s_2} \pmod{n}$$

$$= g^{w(r_1 - c(u-2^{l_1}) - c2^{l_1}) - (r_2 - cuw)} \pmod{n}$$

$$= g^{wr_1 - r_2} \pmod{n}$$

$$= B^{r_1} / g^{r_2} \pmod{n} = d_1$$

$$d_2' = D^{s_4 - c2^{l_1}} / y^{s_5 - c2^{l_1}} \pmod{n}$$

$$= g^{(u+xw)[(r_4 + u + c2^{l_1}) - c2^{l_1}] - x(r_5 + uw + c2^{l_1} - c2^{l_1})} \pmod{n}$$

$$= g^{u^2 + (u+xw)r_4 - xr_5} \pmod{n}$$

$$= g^{u^2} D^{r_4} / y^{r_5} \pmod{n} = d_2$$

$$d_3' = B^c g^{s_3} \pmod{n}$$

$$= g^{wc + r_3 - cw} \pmod{n}$$

$$= g^{r_3} \pmod{n} = d_3$$

$$d_4' = D^c g^{s_1 - c2^{l_1}} y^{s_3} \pmod{n}$$

$$= g^{(u+xw)c + r_1 - c(u-2^{l_1}) - c2^{l_1} + x(r_3 - cw)} \pmod{n}$$

$$= g^{r_1 + xr_3} \pmod{n}$$

$$= g^{r_1} y^{r_3} \pmod{n} = d_4$$

Hence $c' = c$ and ($s_1$, $s_2$, $s_3$, $s_4$, $s_5$) are in their intervals respectively. The verifier would always accept the forged group signature.

Moreover, the group manager cannot recover ID since $(D / B^x) = g^u$ that is not the identity of any group member stored in the group database.

The security weakness comes from the fact that the verification equation does check the membership certification ($A_i$, $X_i$) with the property $A_i^e = C + x_i \pmod{n}$

## 4. Conclusions

We have pointed out that there are some problems in the Popescu group signature scheme. Though we can choose larger primes as $p$ and $q$, which would double the bit size of the composite number $n$. However, this modification would reduce the efficiency of the Popescu group signature scheme. Even so, there still exists

a fatal security problem in the Popescu group signature scheme. Any adversary can easily forge group signatures for any message on behalf of the group. The group manager cannot reveal the identity of the forger, and verifiers cannot find forged group signature.

The security problem results from that the verification equation does not verify the membership certification. Though the membership certification is required to compute group signatures, the adversary can use a false one instead of the valid one since verifiers cannot discover this kind of fraud.

# REFERENCE


1. D. CHAUM, E. HEYST: **Group signature**, Pre-proceeding of Eurocrypt'91, Brighton, Uk, 8-11, pp.257-265
2. G. ATENIESE, J. CAMENISCH, M. JOYE, G. TSUDIK: **A practical and provably secure coalition-resistant group signature scheme**, Advances in Cryptology – CRYPTO 2000, LNCS 1880, Springer, Berlin, pp.255-270
3. J. CAMENISCH: **Efficient and generalized group signatures**, Advances in Cryptology – EUROCRYPTO'97, LNCS 1233, Springer, Berlin, 1997, pp. 465-479
4. L. CHEN, T.P. PEDERSEN: **New group signature schemes**, Advances in Cryptology – EUROCRYPTO'94, LNCS 950, Springer, Berlin, 1995, pp. 171-181
5. J. CAMENISCH, M. STADLER: **Efficient group signature schemes for large groups**, Advances in Cryptology – CRYPTO'97, LNCS 1296, Springer, Berlin, 1997, pp.410-424
6. M. JOYE, S. KIM, N.-Y. LEE: **Cryptanalysis of two group signature schemes**, Proc. Of Information Security Workshop'99, LNCS 1729, Springer, Berlin, 1999, pp.271-275
7. H. PEDERSEN: **How to convert any digital signature scheme into a group signature scheme**, Security Protocols Workshop, LNCS 1361, Springer, Berlin, 1997, pp.177-190
8. W. B. LEE, C. C. CHANG: **Efficient group signature scheme based on the discrete logarithm**, IEE Proc. Comput. Digit. Tech., 1998, 145, pp.15-18
9. C. POPESCU: **An efficient group signature scheme for large groups**, Studies in Informatics and Control Journal, Vol. 10, No. 1, 2001, pp.7-14
10. T. OKAMOTO, A. SHIRAISHI: **A fast signature scheme based on quadratic inequalities**, Proceedings of IEEE symposium on security and privacy, 1995, pp.123-132