Comments on Abe et al.'s Threshold Signer-ambiguous Signature Scheme from Variety of Keys

Ya-Fen Chang [1] **and Chin-Chen Chang** [2, 3]
[1]Department of Management Science,
National Taichung Institute of Technology, Taichung 404, Taiwan, R.O.C.
E-mail: cyf@cs.ccu.edu.tw
[2]Department of Computer Science and Information Engineering,
National Chung Cheng University, Chiayi, Taiwan, 621, R.O.C.
[3]Department of Information Engineering and Computer Science,
Feng Chia University, Taichung, Taiwan, 40724, R.O.C.
E-mail: ccc@cs.ccu.edu.tw

Abstract

In 2004, Abe et al. proposed a threshold signer-ambiguous signature scheme from variety of keys. Their scheme is a generalized case of the ring signature scheme, and it allows the key types to be based the trapdoor one-way permutations (TOWP) or sigma-protocols including Schnorr's signature scheme. However, the signed message is public for all, which may result in disputes. In this paper, we present a novel threshold signer-ambiguous signature scheme, having the signed message concealed and keeping who the receivers are secret from variety of keys.

Keywords: Trapdoor one-way permutation, digital signature, signer-ambiguous signature, ring signature, Schnorr's signature scheme

1. Introduction

For many applications, anonymity is an important issue. As for the digital signature, anonymity could be still amended even though the digital signature is used to authenticate the signer of the corresponding document. In [15], one motivation for the above scenario comes into being. One of the possible signers can sign the document without the other possible signers' agreement when the signed document may be harmful if exposed to be public. Note that the verifiers know the possible signers instead of the real signer to have the document trustworthy. Consequently, the real signer should be ambiguous instead of anonymous. Thus, signer-ambiguous signature schemes are preferred to be setup-free such that the real signer can select the possible signers at will to make himself/herself be able not to be noticed. On the other hand, in the threshold signature schemes [9, 10, 17] and in the group signature schemes [5-7], the possible signers are grouped to be a set after the setup process.

Several proposed schemes [11, 12, 15] can be adopted as setup-free signer-ambiguous signature schemes. The partial knowledge proof CDS [11] leads efficient threshold signer-ambiguous schemes, and it can be combined with other signature schemes based on sigma-protocols such as Schnorr's signature scheme. Nevertheless, the signature schemes based on TOWP cannot be employed in CDS—RSA and Rabin signature schemes [3, 8] for example.

Rivest et al. proposed the ring signature scheme which almost directly adopts TOWP [15]. Bresson et al. proposed a t-out-of-n threshold ring signature scheme with the signature size exponential to the threshold t [4]. Later, a more efficient version was presented such that the signature size is linear to t and n [13]. Meanwhile, Abe et al. presented a modification on the ring signature scheme such that it can be based on both of sigma-protocols and TOWP [1], where the modification is 1-out-of-n. In 2004, Abe et al. proposed a t-out-of-n signer-ambiguous signature scheme [2]. They claimed that the base signature schemes can be based on sigma-protocols including Schnorr's signature scheme, or TOWP. After analyzing Abe et al.'s scheme, we observe that Schnorr's scheme cannot be directly applied to their scheme, and the base signature based on sigma-protocols may be insecure.

2. Preliminaries

In the following, we introduce two types of signature schemes, type-OW and type-3M, which employ TOWP and sigma-protocols, respectively. Type-OW includes schemes such as the variants of RSA signature scheme, Rabin's signature scheme [3, 8] and Paillier's signature scheme [14], which use one-way trapdoor permutations. Let F, a claw-free permutation, be a one-way trapdoor permutation and I be the corresponding inverse function. F and I are both defined over the space C. Let SK and PK be the involved private and public keys, respectively. Suppose that EM is the encoded message, where EM $\in$ C. Then the signature s of EM is I(SK, EM), and EM can be obtained by computing EM = F(PK, s). Note that the verifier may check if EM = F(PK, s) to determine ifr the signature s of EM is valid.

Type-3M, typified by Schnorr's signature scheme, includes schemes derived from the sigma protocols. There are three polynomial-time algorithms A, Z and V performed by the signer and the verifier. The signer commits to a $\leftarrow$ A (SK; r), randomly chooses the challenge c and computes s = Z(SK, r, c). The verifier checks if a = V(PK, c, s) to verify the signature.

3. Abe et al.'s Threshold Signer-ambiguous Signature Scheme

In this section, the details of Abe et al.'s scheme are shown. First of all, the initialization is presented as follows. Let the set of the involved public keys be G = $\{PK_1, PK_2, \ldots, PK_n\}$, where the first v keys of G are of type-OW and the others are of type-3M. At least t corresponding private keys are known to the signers. Let p' be a prime larger than any number in the challenge space C_i determined by $PK_i \in G$ for i = 1, 2,…, n. For i = 1, 2,…, n, let H_0, H_i and K_i be hash functions with the hashing results in $Z_{p'}$, C_i and C_i, respectively. The signature scheme is composed of two phases: the signature generation phase and the verification phase described in Subsections 3.1 and 3.2, respectively. In Subsection 3.3, an example is given.

3.1. The Signature Generation Phase

Suppose that (G, t, m) are given, the corresponding signature α is generated as follows.

Step 1: For the real signer U_i, he/she chooses a_i from C_i if U_i's key is of type-OW or computes $a_i \leftarrow A(SK_i; r_i)$ if U_i's key is of type-3M.

Step 2: For other signer U_i who does not sign m, z_i is randomly chosen from $Z_{p'}$, s_i is chosen from S_i, and c_i and a_i are computed, where S_i is the signature space. If U_i's key is of type-OW, $c_i = H_i(z_i)$ and $a_i = F_i(PK_i, s_i) - c_i$. If U_i's key is of type-3M, $c_i = K_i(z_i)$ and $a_i = V_i(PK_i, c_i\ s_i)$. Note that this step is performed by the real signers.

Step 3: $z_0 = H_0(G, t, m, a_1, a_2, \ldots, a_n)$ is computed, and an (n-t)-degree polynomial P over $Z_{p'}$ is found, where $P(i) = z_i$.

Step 4: For the real signer U_i, he/she computes $c_i = H_i(P(i))$ and $s_i = I_i(SK_i, a_i + c_i)$ if U_i's key is of type-OW, or he/she computes $c_i = K_i(P(i))$ and $s_i = Z_i(SK_i, r_i, c_i)$ if U_i's key is of type-3M.

3.2. The Verification Phase

While given (G, t, m) and the signature $\alpha = (P, s_1, s_2, \ldots, s_n)$, the verifier performs as follows to verify the signature.

Step 1: If U_i's key is of type-OW, the verifier computes $a_i = F_i(PK_i, s_i) - H_i(P(i))$.

Step 2: If U_i's key is of type-3M, the verifier computes $a_i = V_i(PK_i, K_i(P(i)), s_i)$.

Step 3: The verifier checks if $P(0) = H_0(G, t, m, a_1, a_2, \ldots, a_n)$. If it holds, the verifier is convinced that the obtained signature α is valid.

3.3. An Example of Abe et al.'s Scheme

In [2], Abe et al. presented an example of a t-out-of-n signer-ambiguous signature scheme, where RSA and the Schnorr-like signature schemes are applied, t = 2, and n = 4. We extend Abe et al.'s example such that t = 3 and n = 5.

Let G = $\{PK_1, PK_2, PK_3, PK_4, PK_5\}$. The key types for U_1 and U_2 are of RSA signature scheme, and the others are of the Schnorr-like signature scheme. For i = 1 and 2, $(SK_i, PK_i) = (d_i, (n_i, e_i))$, where $e_i \in Z_{\phi(n_i)}$ and $d_i = e_i^{-1} \bmod \phi(n_i)$. For i = 3, 4, 5, $(SK_i, PK_i) = (x_i, (g_i, q_i, p_i, y_i))$, where g_i is the primitive element with the order q_i and the modulus p_i, q_i is a great prime factor of $\phi(p_i)$ and $y_i = g_i^{x_i} \bmod p_i$. Let p' be a prime greater than n_1, n_2, p_3, p_4 and p_5. Let H_0, H_1, H_2, K_3, K_4 and K_5 be hash functions with results in $Z_{p'}$, Z_{n_1}, Z_{n_2}, Z_{q_3}, Z_{q_4}, and Z_{q_5}, respectively.

Suppose that U_1 and U_3 are the real signers who are going to sign the message m. The followings are performed.

Step 1: U_1 chooses a_1 from Z_{n_1}. U_3 computes $a_3 = g_3^{r_3} \bmod p_3$.

Step 2: z_2 is randomly chosen from $Z_{p'}$, s_2 is chosen from Z_{n_2}, and $c_2 = H_2(z_2)$ and $a_2 = (s_2^{e_2} - c_2) \bmod n_2$ are computed. For i = 4, 5, z_i is randomly chosen from $Z_{p'}$, s_i is chosen from Z_{q_i}, $c_i = K_i(z_i)$ and $a_i = g_i^{s_i} y_i^{-c_i} \bmod p_i$ are computed. This step is executed by U_1 and U_3.

Step 3: $z_0 = H_0(G, t, m, a_1, a_2, a_3, a_4, a_n)$ is computed, and a 3-degree polynomial P over $Z_{p'}$ is found, where $P(0)=z_0$, $P(2)=z_2$, $P(4)=z_4$, and $P(5)=z_5$.

Step 4: U_1 computes $c_1 = H_1(P(1))$ and $s_1 = (a_1 + c_1)^{d_1} \bmod n_1$. U_3 computes $c_3 = K_3(P(3))$ and $s_3 = (r_3 + c_3x_3) \bmod q_3$.

Step 5: Finally, the signer-ambiguous signature $\alpha = (P, s_1, s_2, s_3, s_4, s_5)$ is obtained.

When the verifier wants to verify the signature α, he/she performs as follows:

Step 1: The verifier computes $a_i = (s_i^{e_i} - H_i(P(i))) \bmod n_i$ for $i = 1, 2$.

Step 2: The verifier computes $a_i = g_i^{s_i} y_i^{-K_i(P(i))} \bmod p_i$ for $i = 3, 4, 5$.

Step 3: The verifier checks if $P(0) = H_0(G, t, m, s_1, s_2, s_3, s_4, s_5)$. If it holds, the verifier is convinced that the obtained signature α is valid.

4. Schnorr's Signature Scheme

In this section, we review Schnorr's signature scheme [16]. First, two primes, p and q, are chosen, where q is a prime factor of (p-1). Second, a primitive element g is chosen, where $g \neq 1$ and $g^q \bmod p = 1$. Note that g, p, and q are all public. For the user U, he/she chooses the private key x less than q and computes the corresponding public key $y = g^{-x} \bmod p$.

When U wants to sign a message m, he/she performs as follows:

Step 1: Chooses a random number r, less than q, and computes $j = g^r \bmod p$.

Step 2: Computes $a = h(M, j)$, where h() is a one-way hash function.

Step 3: Computes $s = (r + x*a) \bmod q$.

After the three steps, U generates the digital signature (a, s) for M. When the verifier V wants to verify the signature, he/she performs as follows:

Step 1: Computes $t' = g^s * y^a \bmod p$.

Step 2: Computes $a' = h(M, t')$ and checks if a' equals a. If it holds, V confirms the validity of the signature (a, s); otherwise, the received signature is regarded as an illegal one.

5. Discussions

After reviewing Abe et al.'s scheme and Schnorr's signature scheme, we observe that Schnorr's signature scheme cannot be employed in Abe et al.'s proposed scheme because the real signer cannot generate the valid signature for other candidate signers. The details are shown in Subsection 5.1. In Subsection 5.2, we show that the type-3M base signature scheme is insecure. More discussions are given in Subsection 5.3.

5.1. Another Example

Schnorr's signature scheme in Section 4 is employed in the example in Subsection 3.3. Note that $y_i = g_i^{-x_i} \bmod p_i$. The following procedures are performed to generate the signature of the message m.

Step 1: U_1 chooses a_1 from Z_{n_1}. U_3 computes $a_3 = K_3(g_3^{r_3} \bmod p_3, m)$.

Step 2: z_2 is randomly chosen from $Z_{p'}$, s_2 is chosen from Z_{n_2}, and $c_2 = H_2(z_2)$ and $a_2 = (s_2^{e_2} - c_2) \bmod n_2$ are computed. For $i = 4, 5$, z_i is randomly chosen from $Z_{p'}$, s_i is chosen from Z_{q_i}, $c_i = K_i(z_i)$.

However, for $i = 4, 5$, a_i cannot be computed by the real signer on behalf of U_i. The reasons are shown as follows:

Since $s_i = (r_i + x_i*a_i) \bmod q_i$, we have the followings.

$$r_i = (s_i - x_i*a_i) \bmod q_i . \qquad (1)$$

$$g_i^{r_i} = g_i^{s_i} y_i^{a_i} \bmod p_i . \qquad (2)$$

And, because $a_i = K_i(j_i = g_i^{r_i} \bmod p_i, m)$, Equation (2) can be rewritten as follows:

$$j_i = g_i^{s_i} y_i^{K_i(j_i,m)} \bmod p_i . \qquad (3)$$

According to Equation (3), it is observed that j_i cannot be retrieved because of the difficulties of solving the discrete logarithms and the security of the hash function even though s_i, g_i, m and y_i are known. In other words, only the user who knows x_i can generate a_i. According to the above analyses, it is ensured that Schnorr's signature scheme cannot be adopted in Abe et al.'s signer-ambiguous signature scheme.

5.2. Cryptanalysis of Abe et al.'s Scheme

As shown in [2], Abe et al. presented an example. The type-3M base signature scheme in Abe et al.'s example is shown as follows. First, two primes, p and q, are chosen, where q = 2p+1. Then, a primitive element g is chosen, where $g \neq 1$ and $g^q \bmod p = 1$. Note that g, p, and q are all public. The user U, whose private key is x, possesses the corresponding public key $y = g^x \bmod p$. When U wants to sign a message m, he/she performs as follows:

Step 1: Chooses a random number r and computes $a = g^r \bmod p$.

Step 2: Computes $s = (r + x* h(m)) \bmod q$.

After the above two steps, U generates the digital signature (a, s) for m. When the verifier V wants to verify the signature, he/she checks whether $g^s = a * y^{h(m)} \bmod p$ holds or not. If it holds, V ensures the validity of the signature (a, s); otherwise, the received signature is regarded to be illegal.

In the type-3M base signature scheme, the malicious user Eve can impersonate the legal user U to sign the message at will without knowing U's private key x. The details are shown as follows:

Step 1: Eve chooses the desired message M′.

Step 2: Eve randomly chooses $s' \in Z_q$.

Step 3: Eve computes $a' = g^{s'} * (y^{h(M')})^{-1} \bmod p$.

Once the verifier wants to verify the signature (a′, s′) for M′, he/she checks whether $g^{s'} = a' * y^{h(M')}$ mod p holds. Unfortunately, the forged signature must be verified successfully even though U is not the real signer. To sum up, Eve generates the valid signature (a′, s′) for M′ without knowing U's private key x.

5.3. More Discussions

According to the above analyses shown in Subsections 5.1 and 5.2, the secure type-3M base signature scheme should be modified as follows: The signer commits to a ← A (SK; r), randomly chooses the challenge c and computes s = Z(SK, r, c, a). The verifier checks if a = V(PK, c, s, a) to determine the validity of the signature.

Nevertheless, the modified type-3M signature scheme cannot be employed to Abe et al.'s ignature scheme—Schnorr's signature scheme for example. It is because the message cannot be retrieved if the signature is determined at first in the secure signature scheme. Thus, the real signers cannot generate the partial signature for other signers.

6. Conclusions

In 2004, Abe et al. proposed a novel threshold signer-ambiguous signature scheme from variety of keys. They claimed that their scheme allows the base signature schemes to be based on sigma-protocols, including Schnorr's signature scheme, or claw-free permutations. After analyzing the above observations, it is observed that the base signature scheme for the keys of the Schnorr-like signature scheme may be insecure and may be designed only for Abe et al.'s scheme. Thus, it is ensured that the base signatures in Abe at al.'s scheme cannot be any one belonging to sigma-protocol or claw-free permutations.

References

[1] M. Abe, M. Ohkubo, and K. Suzuki, "1-out-of-n Signatures from Variety of Keys," *ASIACRYPT 2002*, Vol. 2501, pp. 415-432, Springer-Verlag, 2002.

[2] M. Abe, M. Ohkubo, and K. Suzuki, "Efficient Threshold Signer-ambiguous Signatures from Variety of Keys," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, Vol. E87-A, No. 2, pp. 471-479, February 2004.

[3] M. Bellare and P. Rogaway, "The Exact Security of Digital Signatures-How to Sign with RSA and Rabin," *EUROCRYPT'96*, Vol. 1070, pp. 399-416, Springer-Verlag, 1996.

[4] E. Bresson, J. Stern, and M. Szydlo, "Threshold Ring Signatures and Applications to Ad-hoc Groups," *CRYPTO'02*, Vol. 2442, pp. 465-480, Springer-Verlag, 2002.

[5] J. Camenisch, "Efficient and Generalized Group Signatures," *EUROCRYPT'97*, Vol. 1233, pp. 465-479, Springer-Verlag, 1997.

[6] J. Camenisch and M. Stadler, "Efficient Group Signature Schemes for Large Groups," *CRYPTO'97*, Vol. 1294, pp. 410-424, Springer-Verlag, 1997.

[7] D. Chaum and E. Van Heyst, "Group Signatures," *EUROCRYPT'91*, Vol. 547, pp. 257-265, Springer-Verlag, 1991.

[8] J. S. Coron, "Optimal Security Proofs for PSS and Other Signature Schemes," *EUROCRYPT'02*, Vol. 2332, pp. 272-287, Springer-Verlag, 2002.

[9] Y. Desmedt and Y. Frankel, "Shared Generation of Authenticators and Signatures," *CRYPTO'91*, Vol. 576, pp. 457-469, Springer-Verlag, 1992.

[10] R. Gennaro, S. Jarecki, H. Krawczyk, and T. Rabin, "Robust Threshold DSS Signature," *EUROCRYPT'96*, Vol. 1070, pp. 354-371, Springer-Verlag, 1996.

[11] R. Gramer, I. Damgard, and B. Schoenmakers, "Proofs of Partial Knowledge and Simplified Design of Witness Hiding Protocols," *CRYPTO'94*, Vol. 1839, pp. 174-187, Springer-Verlag, 1994.

[12] M. Jakobsson, K. Sako, and R. Impagliazzo, "Designated Verifier Proofs and Their Applications," *EUROCRYPT'96*, Vol. 1070, pp. 143-154, Springer-Verlag, 1996.

[13] H. Kuwakado and H. Tanaka, "Digital Signature Schemes with Anonymous Signers," *IPSJ SIGNotes Computer Security*, No. 018-35, 2002.

[14] P. Paillier, "Public-key Cryptosystems Based on Composite Degree Residuosity Classes," *EUROCRYPT'99*, Vol. 1592, pp. 223-238, Springer-Verlag, 1999.

[15] R. Rivest, A. Shamir, and Y. Tauman, "How to Leak a Secret," *ASIACRYPT'01*, Vol. 2248, pp. 552-565, Springer-Verlag, 2001.

[16] C. P. Schnorr, "Efficient Signature Generation by Smart Cards," *Journal of Cryptology*, Vol. 4, No. 3, pp. 161-174, 1991.

[17] V. Shoup, "Practical Threshold Signatures," *EUROCRYPT'00,* Vol. 1807, pp. 207-220, Springer-Verlag, 2000.