Improvements of Threshold Signature and Authenticated Encryption for Group Communications

Min-shiang Hwang¹, Tingyi Chang² and Chengying Yang³*

¹Department of Computer Science & Information Engineering, Asia University, Taiwan;
²Graduate Institute of E-Learning, National Changhua University of Education, Changhua 500, Taiwan;
³Department of Computer Science, University of Taipei, Taipei 100, Taiwan
Email: cyang@utaipei.edu.tw

Abstract. In this paper, we propose an improved threshold signature and authenticated encryption for group communications. The proposed scheme gives the user the power to dynamically designate the participants in the verifying group so that they can cooperate to verify the threshold signature. The improved scheme is more practical in real-world applications.

1. Introduction

Paper work is rapidly being replaced as e-mail, electronic commerce, electronic money, etc. become more and more widespread. In many of these new forms of communication, a digital signature is essential. Digital signatures have become more and more important in modern electronic society because they can offer such properties as integrity and authentication [1-7]. Integrity guarantees that a message being transferred never gets corrupted, and authentication guarantees that the signer cannot be impersonated. Traditional digital signatures, such as RSA [8-11] and DSA [12-13], only allow a single signer to sign a message, and anyone can verify the signature at any time. However, there are growing and growing numbers of times when a message needs to be signed by a set of signers and for distributing the power of a single authority. Multisignature schemes [14-23] and threshold signature schemes [24-37] have thus been designed to solve such problems.

In 2000, Wang et al. [38] proposed a scheme that covers the situation where the documents between business entities need to be signed and verified. That is, the documents will not be exposed to any outsider. For example, when two companies have to communicate with each other, some specified signers may have to sign certain documents according to their positions in the company, and some special verifiers may be assigned to check on these signatures. In addition, there are usually confidential data that need to be encrypted or decrypted by some specified participants. Wang et al. have brought up a new idea that the \((t, n)\) threshold signature on behalf of the signing group should be able to be verified by \((k, l)\) threshold-shared verification on behalf of the verifying group. They have proposed two schemes: the \((t, n)\) threshold signature with \((k, l)\) threshold-shared verification and the \((t, n)\) threshold-authenticated encryption with \((k, l)\) threshold-shared verification.

In their schemes [38], the shared distribution center (SDC) is responsible for dividing the signing group's and verifying group's secret keys into \(n\) and \(l\) different shadows and the associating the groups' and participants' public keys to the individual groups and participants, respectively. By using the Lagrange interpolation formula, \(t\) participants in the signing group and \(k\) participants in the verifying group have the ability to compute a common session key shared between the two groups by using their
shadows and the opposite group's public key. The common session key is used to ensure the communication between the two groups. Any t or more participants in the signing group can use their shadows to generate their individual signatures and hand over these individual signatures to a clerk. Then, the clerk can verify these individual signatures and combine these t valid individual signatures to generate a threshold signature on behalf of the signing group. On the other hand, any k or more participants in the verifying group have the ability to collaborate to verify the threshold signature.

Unfortunately, Hsu et al. [39] and Tseng et al. [40] separately proved that Wang et al.'s schemes are not robust enough against forgery and that something is wrong with the verification of threshold signatures because the common session keys are the same for different threshold signature. Anyone can obtain the signing group's secret key from two valid threshold signatures. They separately proposed improved schemes on Wang et al.'s schemes. Recently, Lee [41] pointed out that the signing group's secret key of Tseng et al.'s improved schemes is also apt to be disclosed. On the other hand, though Hsu et al.'s improved scheme can successfully withstand the attack, their scheme has the following disadvantages in practice.

- The SDC must take part in the generation of each individual signature and threshold signature as well as the distribution of fresh session keys to all the participants.
- Each participant in the signing group should keep two secret keys to sign a message.
- The signing group and the verifying group cannot exchange their roles with each other.
- \((k, l)\) threshold-shared verification is immovable.

In this paper, we will focus on \((t, n)\) threshold-authenticated encryption with \((k, l)\) threshold-shared verification. Our scheme not only can live up to the requirements an ideal \((t, n)\) threshold-authenticated encryption scheme with \((k, l)\) threshold-shared verification should but also can get rid of the disadvantages mentioned above. Hence, our scheme is more practical and efficient in real-world applications than Hsu et al.'s scheme.

Technically, our new scheme combines Harn's threshold signature scheme [42] and the concept of generalization group-oriented cryptosystem [43]. In the generalized group-oriented cryptosystem, a sender sends an encrypted message to a group such that the received message can only be decrypted by the authorized subsets of members in the receiving group. Hence, we use the concept of generalized group-oriented cryptosystem to encrypt the threshold signature generated by the signing group and then enable the verifiers in the verifying group to decrypt and verify the message. However, most of the existing generalized group-oriented cryptosystems [44-45] use an additional asymmetric cryptosystem to encrypt/decrypt the message, which is not suitable for our purposes because there has existed a public-key system in the two groups. In this paper, we shall propose a new generalized group-oriented cryptosystem to integrate the threshold signature scheme.

The remainder of our paper is organized as follows. In Sections 2, we will review the Hsu et al.'s scheme. In Section 3, we will point out the weakness of Hsu et al.'s scheme and propose an improved threshold signature and authenticated encryption for group communications. Finally, the conclusion will be in the last section.

2. Review of Hsu et al.'s Scheme

In this section, we shall first go over Hsu et al.'s scheme and then point out the disadvantages of their scheme. The notation \(G_s = \{u_1, u_2, \ldots, u_n\}\) is defined as the group of n signers and \(G_v = \{v_1, v_2, \ldots, v_l\}\) as the group of l verifiers. Any t out of n signers can act on behalf of the signing group to generate the threshold signature, and any k out of l verifiers can work on behalf of the verifying group to authenticate the signature. Furthermore, any of the signers who cooperate to get the threshold signature can be randomly elected as a specified clerk to authenticate the individual signatures and combine the t valid individual signatures to compute the threshold signature. The scheme is divided into three phases:

1. Parameters generating phase,
2. Individual signature generating and verifying phase, and
3. Threshold signature generating and verifying phase.

The details are as follows:
2.1. Parameters Generating Phase:
In this phase, the SDC is responsible for initializing the system and generating the parameters as follows.

1) Randomly choose two large prime numbers \( p \) and \( q \), where \( q|p-1 \), and a generator \( g \) of order \( q \) in \( \text{GF}(p) \).
2) Randomly choose three secret polynomials over \( \mathbb{Z}_{pq} \), \( f_i(x) = a_{i0} + a_{i1}x + \ldots + a_{ir_i} x^{r_i} \), \( f_b(x) = b_0 + b_1 x + \ldots + b_{r_b} x^{r_b} \), and \( f_s(x) = v_0 + v_1 x + \ldots + v_{r_s} x^{r_s} \), where \( 0 < a_{ij}, b_j, v_j < q \).
3) Compute \( f_i(0) = a_0 \) as a group secret key of \( G_i \). The associated group public key is \( Y_i = g^{a_0} \mod p \).
4) Compute \( f_b(0) = b_0 \) as a group secret value of \( G_b \). The associated group public value is \( Y_b = g^{b_0} \mod p \).
5) Compute \( f_s(0) = v_0 \) as a group secret key of \( G_s \). The associated group public key is \( Y_s = g^{v_0} \mod p \).
6) Randomly choose a public integer \( x_i \in [1,q-1] \) for each \( u_i \) in \( G_i \) and compute their secret key \( f_i(x_i) \) and session key \( f_i(x_i) \), for \( i = 1,2,\ldots,n \). The associated \( u_i \) public key is \( y_i = g^{f_i(x_i)} \mod p \), for \( i = 1,2,\ldots,n \).
7) Randomly choose a public integer \( x_i \in [1,q-1] \) for each user \( u_i \) in \( G_i \) and compute their secret key \( f_i(x_i) \), for \( i = 1,2,\ldots,l \). The associated \( u_i \) public key is \( y_i = g^{f_i(x_i)} \mod p \), for \( i = 1,2,\ldots,l \).

2.2. Individual Signature Generating and Verifying Phase:
In this phase, \( u_i \) generates his/her individual signature, and the clerk verifies the individual signature. Without loss of generality, assume that \( t \) members \( u_1, u_2,\ldots,u_t \) in \( G_t \) are to sign a message \( m \). Each \( u_i \) uses his/her secret key \( f_i(x_i) \), session key \( f_i(x_i) \), and \( G_i \)'s public key \( Y_i \), to compute \( z_i = \left( Y_i \right)^{f_i(x_i)} \mod p \), \( r_i = g^{z_i} \mod p \), where \( L_i = (0-x_i)/(x_{i1}x_{i2}) \) \( \ldots \) \( (0-x_{i1}x_{i2} \ldots x_{i(t-1)}) \mod q \). Then, \( u_i \) transmits \((z_i, r_i)\) to other participants via a secure channel. After receiving the values, \( h/s \) computes \( z = z_{i1}z_{i2} \ldots z_i = g^{z_i} \mod p \), and the individual signature \( s_i = f_i(x_i) L_i - r f_i(x_i) \mod q \). Finally, \( u_i \) sends \( s_i \) to the clerk, who then verifies its validity by checking \( r_i = g^{s_i} \mod p \).

2.3. Threshold Signature Generating and Verifying Phase:
If \( t \) individual signatures are valid, the clerk combines the \( t \) valid individual signatures to compute the threshold signature \( S = s_{i1} + s_{i2} + \ldots + s_t = z f_i(0) - r f_i(0) \mod q \). The threshold signature for message \( m \) is \((r, S)\). Without loss of generality, assume that \( k \) members \( u_1, u_2,\ldots,u_k \) in \( G_k \) are to recover the message \( m \) from \((r, S)\). Each \( u_i \) uses his/her secret key \( f_i(x_i) \) and \( G_i \)'s public key \( Y_i \), to compute \( A = g^S Y_i \mod p \), \( z_i = (Y_i Y_s)^{f_i(x_i)} \mod p \), where \( L_i = (0-x_i)/(x_{i1}x_{i2}) \) \( \ldots \) \( (0-x_{i1}x_{i2} \ldots x_{i(t-1)}) \mod q \). Then, \( u_i \) transmits \( z_i \) to other participants via a secure channel. After receiving the values, \( h/s \) computes \( z = z_{i1}z_{i2} \ldots z_k = \left( g^{z_{i1}z_{i2} \ldots z_k} \right) \mod p \) and recovers \( m = r A \mod p \). Finally, the recovered message \( m \) can be authenticated by checking the validity of the redundancy with it.

3. Improvement of Threshold Signature andAuthenticated Encryption for Group Communications
Note that the secret polynomial \( f_i(x) \) used in Hsu's scheme must be distinct for each individual signature and threshold signature. Otherwise, \( u_i \)'s secret key \( f_i(x_i) \) and \( G_i \)'s secret key \( f_i(0) \) can be computed from two valid individual signatures and threshold signatures, respectively. Assume that \((r_1, s_1)\) and \((r_2, s_2)\) are the valid individual signatures separately for messages \( m_1 \) and \( m_2 \). Then, \( u_i \)'s secret key \( f_i(x_i) \) can be computed as follows:
\[
s_1 = z f_i(x_i) L_{i1} - r_1 f_i(x_i) L_{i2} \mod q,
\]
\[
s_2 = z f_i(x_i) L_{i1} - r_2 f_i(x_i) L_{i2} \mod q,
\]
Since the value of \( z f_i(x_i) \) for different individual signatures is the same, anyone can compute \( u_i \)'s secret key \( f_i(x_i) \) by Equation (1). On the other hand, assume that \((r_1, s_1)\) and \((r_2, s_2)\) are the valid
threshold signatures separately for messages $m_1$ and $m_2$. $G_i$'s secret key $f_i(0)$ can be computed as follows:

$$s_1 = z f_i(0) - r_1 f_i(0) \mod q,$$
$$s_2 = z f_i(0) - r_2 f_i(0) \mod q.$$  (2)

For the same reason, anyone can compute $G_i$'s secret key $f_i(0)$ by Equation (2).

Hence, the SDC must replace the secret polynomial $f_i(x)$ with a new one and distribute fresh session keys $f_i(x)$ over a secure channel to all the participants in the signing group and then republish a value $Y_s$. Furthermore, because the participants in the signing group and verifying group have different numbers of keys, the two groups cannot exchange their roles with each other unless the SDC distributes another session key via a secret channel to all the participants in the verifying group.

4. Conclusion

In this paper, we have shown that the disadvantages in Hsu et al.'s scheme. We also propose an improved threshold signature and authenticated encryption for group communications. By eliminating the disadvantages in Hsu et al.'s scheme, our scheme is more practical in real-world applications than their scheme.

5. Acknowledgements

This work was partially supported by the Ministry of Science and Technology, Taiwan, R.O.C., under contract no.: MOST 106-3114-E-005-001, MOST 106-2221-E-468-002, and MOST 106-2221-E-845-001.

6. References

[1] Krishna A V N, Narayana A H and Murthy S K 2017 Int. J. Electron. Inf. Engineering 6 87
[2] Zhang Y, Li H, Li X and Zhu H 2015 Int. J. Electron. Inf. Engineering 3 19
[3] Srinath M S and Chandrasekaran V 2018 Int. J. Netw. Secur. 20 9
[4] Ren F, Zheng D and Wang W J 2017 Int. J. Netw. Secur. 19 1072
[5] Liu D, Zhang S, Zhong H, Shi R and Wang Y 2017 Int. J. Netw. Secur. 19 127
[6] Chen Y, Zhao Y, Hu X and Feng Y 2017 Int. J. Netw. Secur. 19 573
[7] Tsai C Y, Ho P F and Hwang M S 2018 Int. J. Netw. Secur. 20 201
[8] Changchien S W, Hwang M S and Hwang K F 2002 Int. J. Computational and Numerical Analysis and Applications 2 303
[9] Sharma G, Bala S and Verma A K 2016 Int. J. Netw. Secur. 18 82
[10] Chang C C, Sun C Y and Chang S C 2016 Int. J. Netw. Secur. 18 201
[11] Deng L, Huang H and Qu Y 2017 Int. J. Netw. Secur. 19 229
[12] Hwang M S, Lee C C and Lu E J L 2001 Pakistan J. Appl. Sci. 1 287
[13] Khan A U and Ratha B K 2017 Int. J. Netw. Secur. 19 599
[14] Chande M K, Lee C C and Li C T 2017 Int. J. Netw. Secur. 19 340
[15] Chow S M 2008 Int. J. Netw. Secur. 7 348
[16] Ahmad R, Deng L, Zhang Y and Zeng J 2018 Int. J. Netw. Secur. 20 403
[17] Hwang M S and Lee C C 2005 Int. J. Netw. Secur. 1 1
[18] Lu R, Cao Z and Shao J 2007 Int. J. Netw. Secur. 4 248
[19] Pan C, Li S, Zhu Q, Wang C and Zhang M 2015 Int. J. Netw. Secur. 17 29
[20] Tiwari N and Padhye S 2015 Int. J. Netw. Secur. 17 736
[21] Wang F, Chang C C, Lin C and Chang S C 2016 Int. J. Netw. Secur. 18 90
[22] Zhang J and Zou W 2007 Int. J. Netw. Secur. 5 62
[23] Zhou C X 2015 Int. J. Netw. Secur. 17 340
[24] Chang T Y, Hwang M S and Yang W P 2005 Appl. Math. and Computation 163 169
[25] Chang T Y, Yang C C and Hwang M S 2004 IEE Proc. Commun. 151 179
[26] Chow S M, Go H W, Hui L C K and Yiu S M 2008 Int. J. Netw. Secur. 7 397
[27] Duan Q and Virendra M 2013 Int. J. Netw. Secur. 15 205
[28] Hwang M S and Chang T Y 2005 Int. J. Netw. Secur. 1 123
[29] Hwang M S, Chang T Y and Yang C C 2004 Future Generation Computer Syst. 20 1013
[30] Hwang M S, Lu E J L and Lin I C 2003 IEEE Trans. Knowledge and Data Eng. 15 1552
[31] Mashhadi S 2013 Int. J. Netw. Secur. 15 274
[32] Shao Z 2005 Int. J. Netw. Secur. 1 24
[33] Shao Z 2006 Int. J. Netw. Secur. 3 286
[34] Shao Z 2008 Int. J. Netw. Secur. 7 218
[35] Tzeng S F, Hwang M S and Yang C Y 2004 Computers & Secur. 23 174
[36] Tzeng S F, Yang C Y and Hwang M S 2004 Future Generation Computer Syst. 20 887
[37] Yang C Y, Tzeng S F and Hwang M S 2004 J. Syst. Softw. 73 507
[38] Wang C T, Chang C C and Lin C H 2000 IEICE Trans. Fundamentals E83-A 1228
[39] Hsu C L, Wu T S and Wu T C 2002 Inf. Process. Lett. 81 41
[40] Tseng Y M, Jan J K and Chien H Y 2001 IEICE Trans. Fundamentals E84-A 2606
[41] Lee N Y 2002 IEICE Trans. Fundamentals E85-A 2364
[42] Harn L and Wang F 2016 Int. J. Netw. Secur. 18 710
[43] Lee C C, Chang T Y and Hwang M S 2010 J. Inf. Assurance and Secur. 5 54
[44] Yang C C, Chang T Y and Hwang M S 2013 Inf. Technology and Control 42 61
[45] Yang C C, Chang T Y, Li J W and Hwang M S 2003 Informatica 14 111