A New Multi-Proxy Multi-Signature Scheme

新多人授權予多人的代理簽章法

Hwang, Shin-Jia
Department of Computer Science and Information Engineering, TamKang University, Tamsui, Taipei Hsien, 251, Taiwan, R.O.C.
E-mail: sjhwang@mail.tku.edu.tw

Chen, Chiu-Chin
Department of Information Management, Chaoyang University of Technology
Wufeng, Taichung Country, 413, Taiwan, R.O.C.
E-mail: s8914604@mail.cyut.edu.tw

Abstract

In this paper, a new multi-proxy multi-signature scheme, which is a new kind of proxy signature scheme, is proposed. In this scheme, an original group of signers can authorize a group of proxy signers under the agreement of all singers both in the original group and proxy group. Then only the cooperation of all singers in proxy group could generate multi-proxy multi-signatures. The size of the proxy certificate and the multi-proxy multi-signature is independent on the number of original or proxy singers. The verification of multi-proxy multi-signatures is similar to that of proxy signatures. So the new scheme is efficient. The new scheme also provides the fair protection for the original signer group and the proxy group. Further, there is no secure channel in the new scheme. This new scheme is secure against the insider attack that is a powerful attack on the multisignature schemes.

Keywords: Proxy signatures, multi-proxy signatures, proxy multi-signatures, digital signatures

1.Introduction

In the digital information world, it is important to provide the authenticity and integrity of digital documents. These functions are provided by digital signature schemes. However, digital signature schemes do not provide the proxy function. For the proxy function, Mambo, Usuda, and Okamoto proposed the proxy signature scheme in 1996 [10, 11]. In the proxy signature scheme, any singer, called an original signer, is allowed to authorize a designated person as his proxy signer. Then the proxy signer is able to sign on behalf of an original signer. Since then, many proxy signature schemes were proposed [3-8, 10-21].

There are several kinds of proxy signature schemes. The threshold proxy signature schemes were proposed [4, 15, 18, 21]. In a (t, n) threshold proxy signature scheme, the original signer can authorize a proxy group with n proxy members. Only the cooperation of t or more proxy members is allowed to generate the proxy signatures.

The multi-proxy signature scheme was first proposed in [4]. The multi-proxy signature scheme is a special case of the threshold proxy signature scheme. The

multi-proxy signature scheme allows an original signer to authorize a group of proxy members. Only the cooperation of all the proxy members can generate the multi-signature on behalf of the original signer. In 2000, Yi et al. first proposed the proxy multi-signature schemes [20]. Then some proxy mulit-signature schemes were proposed [3,14]. In a proxy multi-signature scheme, an original signer group can authorize a proxy signer on behalf of the original signer group.

In this paper, a new kind of proxy signature scheme, multi-proxy multi-signature schemes, will be proposed. In the multi-proxy multi-signature scheme, only the cooperation of all members in the original group can authorize a proxy group. Only the cooperation of all members in the authorized proxy group could sign messages on behalf of the original group. In our real life, there exist many applications of multi-proxy multi-signature schemes. For example, for a large building, there are some conflict among the constructors and the householders. All householders of the large building want to depute a lawyer group as their agents. So a group of lawyers are authorized to act on behalf of all householders.

Be inspired of the simple multi-proxy signature scheme [4] and the proxy multi-signature scheme [3], a new multi-proxy multi-signature scheme will be proposed in the next section. In Section 3, the performance and security analysis of our scheme is given. Section 4 is our conclusion.

2. A New Multi-Proxy Multi-signature Scheme

Let p and q be two large prime numbers such that q|(p-1). The public parameter g is a generator with order q in Z_p. Let the original group consist of n original signers U_1, U_2, …, and U_n. The original signer U_i owns their private key x_{ui} and their public key $y_{ui}= g^{x_{ui}} \bmod p$, for i= 1, 2, .., n. Let the proxy group consist of m proxy signers P_1, P_2, …, and P_m. The proxy signer P_j owns their private key x_{pj} and their public key $y_{pj}= g^{x_{pj}} \bmod p$, for j= 1, 2, …, m. The function h is a public one-way hash function. The proxy warrant w specifies the proxy details. The proxy warrant also includes the identities ID_{ui}'s and ID_{pj}'s, the certified public keys y_{ui}'s of the original signers, and the certified public keys y_{pj}'s of the proxy signers. Our multi-proxy multi-signature scheme is divided into three phases: The proxy certificate generation phase, the multi-proxy multi-signature generation phase, and the multi-proxy multi-signature verification phase.

[The Proxy Certificate Generation Phase]

In this phase, all of the proxy signers P_1 ,P_2, …, P_m, and original signers U_1, U_2, …, U_n, cooperate to generate the proxy certificate. They execute the following steps.

Step 1: Each original singer U_i selects a random integer $k_{ui} \in Z^*_q$, computes $K_{ui}= g^{k_{ui}} \bmod p$, and broadcasts his K_{ui} to the other n-1 original signers and m proxy signers, for i= 1, 2, …, n. At the same time each proxy signer P_j also selects a random integer $k_{pj} \in Z^*_q$, computes $K_{pj}= g^{k_{pj}} \bmod p$, and broadcasts his K_{pj} to the other n original signers and m-1 proxy signers, for j= 1, 2, …, m. Here Z^*_q denotes the set {1, 2, …., q}.

Step 2: Each signer U_i (or P_j) computes

$$K=\prod_{i=1}^{n} K_{ui} \prod_{j=1}^{m} K_{pj} \bmod p.$$

Step 3: Each original signer U_i computes $v_{ui}= h(w)x_{ui}y_{ui}+ k_{ui}K \bmod q$ and broadcasts v_{ui} to the other n+m-1 signers. Each proxy signers P_j also computes $v_{pj}= h(w)x_{pj}y_{pj}+ k_{pj}K \bmod q$ and broadcasts v_{pi} to the other n+m-1 signers.

Step 4: Each signer verifies the correctness of v_{ui} by the equation $g^{v_{ui}} \equiv y_{ui}^{y_{ui}\, h(w)} K_{ui}^{K} \pmod p$, for i= 1, 2, …, n, and v_{pj} by the equation $g^{v_{pj}} \equiv y_{pj}^{y_{pj}\, h(w)} K_{pj}^{K} \pmod p$, for j= 1, 2, …, m.

Step 5: Once all of the above equations hold, each member of the proxy group P_j computes $V=\sum_{i=1}^{n} v_{ui} + \sum_{j=1}^{m} v_{pj} \bmod q$, for j= 1, 2, …, m.

Finally, the m proxy signers $P_1, P_2, \ldots, P_m$ are authorized to act for the agent of the n original signers. The proxy certificate is (K, V). It is important that not only n original signers but also m proxy signers reach an agreement to authorize the signers $P_1, P_2, \ldots, P_m$ as proxy signers.

[The Multi-Proxy Multi-signature Generation Phase]

Suppose the proxy group wants to sign a message M on behalf of the n original signers.

Step 1: Each proxy signer P_j randomly selects an integer $t_j \in Z_q^*$, for j= 1, 2, …m.

Step 2: Each proxy signer P_j computes $r_j= g^{t_j} \bmod p$ and broadcasts r_j to the other m-1 proxy signers, for j= 1, 2, …, m.

Step 3: Each proxy signer P_j computes $R=\prod_{j=1}^{m} r_j \bmod p$ and finds s_j satisfying $s_j= (Vt_j+ x_{pj}y_{pj}R)h(M)^{-1} \bmod q$. Finally the individual proxy signature of the message m is (r_j, s_j), for j= 1, 2, …, m.

Step 4: Each proxy signer P_j sends (w, (K, V), M, (r_j, s_j)) to the clerk C, for j= 1, 2, …, m.

Step 5: The clerk C first checks the proxy certificate by the equation $g^{V} \equiv K^{K}[\prod_{i=1}^{n}(y_{ui}^{\,y_{ui}})\prod_{j=1}^{m}(y_{pj}^{\,y_{pj}})]^{h(w)} \pmod p$. If the equation holds, then the clerk C continues the next step. Otherwise C rejects the proxy certificate.

Step 6: The clerk C computes $R= \prod_{j=1}^{m} r_j \bmod p$ and verifies the individual proxy signatures (r_j, s_j)'s by the equation $g^{h(M)s_j} \equiv (r_j)^{V}(y_{pj})^{Ry_{pj}} \pmod p$, for j= 1, 2, …, m. Once all individual proxy signatures are correct, the multi-proxy multi-signature of message m can be generated as (w, (K, V), M, (R, S)) by computing $S=\sum_{j=1}^{m} s_j \bmod q$.

[The Multi-Proxy Multi-signature Verification Phase]

After receiving the multi-proxy multi-signature (w, (K, V), M, (R, S)), the verifier B verifies the multi-proxy multi-signature in two steps. In Step 1, by using the warrant w and the certificate (K, V), the verifier B first checks whether or not the m proxy signers are authorized by the n original signers. Then the verifier B checks the

correctness of the multi-proxy multi-signature (R, S) in Step 2.

Step 1: Verify the warrant w and the certificate (K, V) by the equation $g^V \equiv K^K [\prod_{i=1}^{n} (y_{ui}^{y_{ui}}) \prod_{j=1}^{m} (y_{pj}^{y_{pj}})]^{h(w)}$ (mod p). If the certificate (K, V) is incorrect, then reject the multi-proxy multi-signature (R, S).

Step 2: Check the correctness of the multi-proxy multi-signature (R, S) by $g^{h(M)S} \equiv R^V [\prod_{j=1}^{m} y_{pj}^{y_{pj}}]^R$ (mod p).

3. Security and Performance Analysis

The security and performance analysis of our proposed scheme is given in this session. In essence, the security of our multi-proxy multi-signature scheme is based on the security of the underlying mutisignature scheme. The security basis of the underlying multisignature scheme is the discrete logarithm problem. To reveal the secret key of any signer from his public key is protected by the discrete logarithm problem. The security of the multisignature is also guaranteed by the difficulty of the discrete logarithm problem. Therefore, the secret key of each signer is secure while the multisignature cannot be forged.

Let us consider the security of the multisignatures for the proxy certificates or multi-proxy multi-signatures. The case of proxy certificate (K, V) is considered first. The individual proxy certificate (K_{ui}, v_{ui}) cannot be forged. Without losing the generality, suppose that someone wants to forge the individual proxy certificate (K_{un}, v_{un}). The forger must generates a forged individual certificate (K'_{un}, v'_{un}) passing the verification equation $g^{v'_{un}} \equiv y_{un}^{h(w)y_{un}} K'_{un}^{K'}$ (mod p), where $K' = K_{u1} \times K_{u2} \times \ldots \times K_{u,n-1} \times K'_{u,n} \times K_{p1} \times K_{p2} \times \ldots \times K_{pm}$. If the value of v'_{un} is determined first, he has to solve the equation $K_{un}^{K_{un}} \equiv [g^{v_{un}} (y_{un}^{h(w)y_{un}})^{-1}]^{K_{u1}^{-1} K_{u2}^{-1} \cdots K_{u,n-1}^{-1} K_{p1}^{-1} K_{p2}^{-1} \cdots K_{pm}^{-1}}$ (mod p). According to [2], to find the value of K'_{un} is an extremely difficult problem. If the value of K'_{un} is determined first, to derive v'_{un} form $g^{v'_{un}} \equiv y_{un}^{h(w)y_{un}} K'_{un}^{K'}$ (mod p) is a discrete logarithm problem. By the same reason, the individual certificates (K'_{pj}, v'_{pj})'s are also unforged. Therefore, the proxy certificate cannot be forged for the same reason. For the case of the multi-proxy multi-signatures, by the similar security analysis, we can find that the multi-proxy multi-signatures are also unforged.

The insider attack [9] is considered since it is a powerful attack on the proposed multisignature schemes. To perform the insider attack, any original signer or proxy signer has to change his public key after the public keys of the other singers have been determined. Without losing the generality, suppose that the signer P_m is the malicious signer. He selects an integer a as his secret key. Then he has to make his public key as y' satisfying the equation $g^a \equiv y'^{y'} (\prod_{i=1}^{n} (y_{ui})^{y_{ui}} \prod_{j=1}^{m-1} (y_{pj})^{y_{pj}})$ (mod p). After obtaining the other signers' public keys, he has to compute the value of y' satisfying $y'^{y'} \equiv [\prod_{i=1}^{n} (y_{ui})^{y_{ui}}]^{-1} [\prod_{j=1}^{m-1} (y_{pj})^{y_{pj}}]^{-1} g^a$ (mod p). If the signer fixed the integer y', he will find that he has to solve the discrete logarithm problem to find the value of a. If the signer determines the integer a first, he has to obtain the value of y' by

solving the difficult problem in [2]. Therefore, the insider attack cannot work to forge the proxy certificate. By the similar analysis, the multi-proxy multi-signatures cannot be forged by the insider attack for the equation $y'^{y'} \equiv [\prod_{j=1}^{m-1} (y_{pj})^{y_{pj}}]^{-1} g^{a} \pmod{p}$. Therefore, both the proxy certificates and multi-proxy multi-signatures are secure.

The proxy certificate must be generated by the cooperation of the original group and the proxy group while the multi-proxy multi-signature has to be generated by the agreement of all members in the proxy group. The certificate verification equation $g^{V} \equiv K^{K}[\prod_{i=1}^{n} (y_{ui}{}^{y_{ui}}) \prod_{j=1}^{m} (y_{pj}^{y_{pj}})]^{h(w)} \pmod{p}$ uses the public keys of all original signers and all proxy singers. Since the insider attack cannot work for our scheme, no signer is able to create the proxy certificate or mulit-proxy multisiganture alone. So the proxy certificate must be generated by the cooperation of the original signers and proxy signers. With the same analysis on the multi-signature verification equation $g^{h(m)S} \equiv R^{V}[\prod_{j=1}^{m} y_{pj}^{y_{pj}}]^{R} \pmod{p}$, all proxy signers must be in agreement on the multi-proxy multi-signature generation.

Our proposed scheme supports the fair protection for the proxy group and the original group. Since no one can forge the proxy certificate without the cooperation of the proxy and original groups, no one can generate the multi-proxy multi-signature without the authorization of the original group. On the other hand, the proxy singers' secret keys are used to generate the multi-proxy multi-signature, so no one can forge the multi-proxy multi-signature without the agreement of all members in the proxy group.

Our scheme satisfies the distinguishability and identifiability conditions [10, 11]. No one can forge the multi-proxy multisignture even if he is an original signer. Moreover, the multi-proxy multi-signture is verified by the public keys of all proxy signers. Therefore, the multi-proxy multi-signature generated by the proxy group can be distinguished. Moreover, the proxy singers' certificated public keys are used, it is to identified by the warrant w. On the other hand, the multi-signature generated by the original group can be also identified and distinguished.

The performance analysis of our scheme is given in the following. To briefly express the computation and the communication costs, some symbols are defined. The symbol T_m means the time to execute one modular multiplication. The symbol T_e is the time to execute one modular exponentiation, and the symbol T_h is the time to execute one one-way hash function h. The symbol T_{INV} means the time to execute one modular inverse operation. The time to execute one modular addition or subtraction is neglected since the cost of them is much less than T_m, T_e, or T_{INV}. The symbol $|T|$ is the size of an integer T.

In our scheme, the generation cost for the proxy certificate is given in the following. The computation and communication costs to produce the integer K are $(n+m)T_e + (n+m-1)(n+m)T_m$ and $(n+m-1)(n+m)|p|$,

respectively. The computation cost for the individual proxy certificates are $2(n+m)T_m+(n+m)T_h$ since $x_{ui}y_{ui}$'s and $x_{pj}y_{pj}$'s can be computed in advance. The communication cost for the individual proxy certificates are $(n+m-1)(n+m)|q|$. The computation cost for checking individual proxy certificates is $(n+m-1)(n+m)(3T_e+T_m+T_h)$ since $y_{ui}^{y_{ui}}$'s and $y_{pj}^{y_{pj}}$'s can be computed in advance. The total computation and communication costs to produce the proxy certificate are $(n+m)(3n+3m-2)T_e+2(n+m)^2T_m+(n+m)^2T_h$ and $(n+m-1)(n+m)(|p|+|q|)$, respectively.

In our scheme, the generation cost of one multi-proxy multi-signature is given. The computation cost for the integer R is $mT_e+m(m-1)T_m$ while the communication cost to broadcast R needs $m(m-1)|p|$. The individual multi-proxy multi-signatures' computation cost is $m(3T_m+T_h+T_{INV})$ since $x_{pj}y_{pj}$'s can be computed while the communication cost for sending $(w, (K, V), M, (r_j, s_j))$'s to clerks $m(2|p|+2|q|+|M|+|w|)$. The computation cost for the clerk C checking proxy certificate and individual multi-proxy multi-signatures is $(3m+3)T_e+(2m+1)T_m+(m+1)T_h$ since $[\prod_{i=1}^{n}(y_{ui}^{y_{ui}})\prod_{j=1}^{m}(y_{pj}^{y_{pj}})]$ and $y_{pj}^{y_{pj}}$'s can be precomputed. Therefore, the total computation and communication costs to produce one multi-proxy multi-signature are $(4m+3)T_e+(m^2+4m+1)T_m+(2m+1)T_h+mT_{INV}$ and $m[(m+1)|p|+2|q|+|M|+|w|]$, respectively.

Finally, in our scheme, the verification cost of one multi-proxy multi-signature (w, (K, V), M, (R, S)) is the double cost of the verification of a single multi-signature. Here the group public keys $[\prod_{i=1}^{n}(y_{ui}^{y_{ui}})\prod_{j=1}^{m}(y_{pj}^{y_{pj}})]$ mod p and $[\prod_{j=1}^{m}y_{pj}^{y_{pj}}]$ mod p are precomputed. For our scheme, the verification of the multi-proxy multi-signature is efficient.

4. Conclusions

The new multi-proxy multi-signature scheme brings out the following advantages. The size of the proxy certificate is independent of the numbers of the original signers while the multi-proxy multi-signature is also independent of the numbers of the proxy members. Our scheme does not need secure channels. Our new scheme also provides the fair protection for the original signer group and the proxy group. Moreover, the new scheme provides the distinguishability and identifiability functions. The new scheme is secure against the insider attack [9] which is a powerful attack on multisignature schemes [1]. Finally the verification of our scheme is efficient.

References

[1] Harn, L. (1999): "Digital multisignature with distinguished signing authorities," ELECTRONICS LETTERS, 18th February 1999 Vol. 35 No. 4, pp.294-295.

[2] Harn, L. (1994): "Group-oriented (t, n) threshold digital signature scheme and digital multisignature," IEE Proceedings: Computers and Digital Techniques, Vol. 141, No. 5, Sept 1994, pp. 307-313.

[3] Hwang, S. J. and Chen, Chiu-Chin (2001): "A New Proxy Multi-Signature Scheme," to appear in International Workshop on Cryptology and Network Security,

Tamkang University, Taipei, Taiwan, Sep. 26-28, 2001.

[4] Hwang, S. J. and Shi, Chi-Hwai (2000): "A Simple Multi-Proxy Signature Scheme," Proceedings of the Tenth National Conference on Information Security, Taiwan, 2000, pp. 134-138.

[5] Hwang, S. J. and Shi, Chi-Hwai (2000): "A Proxy Signature Scheme without Using One-Way Hash Functions", 2000 International Computer Symposium, Chiayi, Taiwan, R.O.C., Dec. 6-8, 2000, pp. 60-64.

[6] Hwang, S. J. and Shi, Chi-Hwai (1999): "The Specifiable Proxy Signature," National Computer symposium 1999, Vol. 1334, Taiwan, December 1999, pp. 190-197.

[7] Kim, S., Park, S., and won, D. (1997): "Proxy Signatures, revisited," ICICS '97, Lecture Notes in Computer Science, Vol. 1334, Springer, Berlin, 1997, pp. 223-232.

[8] Lee, Narn-Yih, Hwang, Tzonelih, and Wang, Chin Hung (1998): "On Zhang's Nonrepudiable Proxy Signature Schemes," Third Australasian Conference, ACISP '98, 1998, pp. 415-422.

[9] Li, Z. C., Hui, L. C. K., Chow, K. P., Chong, C. F., Tsang, H. H., and Chan, H. W. (2000): "Cryptanalysis of Harn Digital Multisignature Scheme with Distinguished Signing Authorities," Electronics Letters, Vol. 36, No. 4, 2000, pp. 314- 315.

[10] MAMBO, Masahiro, USUDA Keisuke, and OKAMOTO, Eiji (1996): "Proxy signatures: Delegation of the Power to Sign Message," IEICE. Trans. Fundamentals, E79-A, 9, 1996, pp. 1338-1354.

[11] MAMBO, Masahiro, USUDA, Keisuke, and OKAMOTO, Eiji (1996): "Proxy Signatures for Delegation Signing Operation," Proc. 3nd ACM Conference on Computer and Communication Security, 1996, pp. 48-57.

[12] Seungjoo Kim, Sangjoon Park, and Dongho Won (1997): "Proxy Signatures, Revisited," Information and Communications Security, Beijing, China, November 11-14, 1997, pp. 223-232.

[13] Sun, Hung-Min (2000): "Design of time-stamped proxy signatures with traceable receivers," IEE Proc.-Comput. Digit. Tech, Vol. 147, No. 6, November 2000.

[14] Sun, Hung-Min (2000): "On Proxy (Multi-) Signature Schemes," 2000 International Computer Symposium, Chiayi, Taiwan, R.O.C., Dec. 6-8, 2000, pp. 65-72.

[15] Sun, Hung-Min (1999): "An Efficient Nonrepudiable Threshold Proxy Signature Scheme with Known Signers," Computer Communications, Vol. 22, 1999, pp. 717-722.

[16] Sun, Hung-Min, Hsieh, and Bin-Tsan (1999): "Time-Stamp Proxy Signatures with Traceable Receivers," Proceedings of the Ninth National Conference on Information Security, Taiwan, 1999, pp. 247-253.

[17] Sun, Hung-Min, and Hsieh, Bin-Tsan (1999): "Remark on Two Nonrepudiable Proxy Signature Schemes," Proceedings

of the Ninth National Conference on Information Security, Taiwan, 1999, pp. 241-246.

[18] Sun, Hung-Min, Lee N.-Y., and Hwang, T (1999): "Threshold Proxy Signatures," IEE Proceedings-computers & Digital Techniques, Vol. 146, No. 5, September 1999, pp. 259-263.

[19] Yen, Sung-Ming, Hung, Chung-Pei, and Lee, Yi-Yuan(2000): "Remarks on Some Proxy Signature Schemes", 2000 International Computer Symposium, Chiayi, Taiwan, R.O.C., Dec. 6-8, 2000, pp. 54-59.

[20] Yi, L. Bai, G., and Xiao, G. (2000): "Proxy multi-signature scheme: A new type of proxy signature scheme," Electronics Letters, Vol. 36, No. 6, 2000, pp.527-528.

[21] Zhang, K. (1997): "Threshold Proxy Signature Schemes," 1997 Information Security Workshop, Japan, September 1997, pp. 191-197.