# Security Analysis and Improvement of Hsu et al.'s Threshold Proxy Signature Scheme

Lifeng Guo[1,2], Xiangguo Cheng[2,3], Yang Liu[2,4]
*(Corresponding author: Lifeng Guo)*

Institute of Systems Science, Academy of Mathematics and System Sciences, Chinese Academy of Sciences[1]
Graduate School of Chinese Academy of Sciences, Beijing 100080, P.R. China (Email: lfguo@amss.ac.cn)
Infocommm Security Department, Institute for Infocomm Research ($I^2R$)[2]
21 Heng Mui Keng Terrace, Singapore 119613 (Email: {stulguo,stuyl2}@i2r.a-star.edu.sg)
State Key Laboratory of Integrated Services Network, Xidian University[3]
Xi'an, 710071, P.R.China
Software Institute, School of Electronic Engineering and Computer Science, Peking University[4]
100871, P.R.China (Email: liuyang@cs.pku.edu.cn)

*(Received July 1, 2005; revised and accepted Aug. 2, 2005)*

## Abstract

In 1999, Sun et al. proposed a new $(t, n)$ threshold proxy signature scheme based on Zhang's threshold proxy signature scheme. But in 2003 Hsu et al. pointed out that Sun's scheme suffered from a drawback and demonstrated an improvement to counter it. In this paper we point out that Hsu's scheme suffers from an *insider attack* against their scheme. That is, a malicious proxy signer can forge a valid threshold proxy signature on *any* message. To thwart this attack, some improvements are further proposed.

*Keywords: Coalition attack, insider attack, proxy signature, threshold proxy signature*

## 1 Introduction

In 1996, Mambo et al. [9, 10] first introduced the concept of the proxy signature schemes. A proxy signature scheme allows an original signer to delegate his/her signing capability to a person, called a proxy signer, to sign on his/her behalf. Up to now, proxy signatures have been widely discussed and suggested for use in many applications, particularly in distributed computing where delegation of rights is quite common. Examples include e-cash systems [11], mobile agents for electronic commerce [6, 7], mobile communication [12], grid computing [3], global distribution networks [1], and distributed shared object systems [8].

For the group-oriented applications, threshold proxy signatures have been proposed. The threshold proxy signature schemes have been widely studied recently [5, 15, 16]. Specifically, a $(t, n)$ threshold proxy signature scheme is a variant of usual proxy signatures in which the proxy signature key is now shared by a group of $n$ proxy signers in such a way that any $t$ or more proxy signers can cooperatively employ the proxy signature key to sign messages on behalf of an original signer, but $t-1$ or fewer proxy signers cannot. In 1999, Sun first introduced a threshold proxy signature scheme [13] based on Zhang's threshold proxy signature scheme. But Hsu et al.'s scheme [4] proposed that Sun et al.'s signature scheme suffered from a weakness, that the proxy signers might change the threshold value. That is, the proxy signers in the designated proxy group can arbitrarily modify the threshold strategy without being detected by the original signer or verifiers, which might violate the original signer's intent. To defeat the weakness, Hsu et al. proposed an improvement of Sun et al.'s threshold proxy signature scheme. However, in this paper we successfully identify an *insider attack* against their scheme. That is, this attack allow a malicious proxy signer to forge a valid proxy signature on *any* given message. Moreover, we point out why their security argument for internal attacks is incorrect. To thwart this attack, some effective improvements are also proposed.

The rest of this paper is organized as follows. Section 2 reviews the Hsu's threshold proxy scheme and Section 3 we demonstrates our security analysis on this scheme. Furthermore, in Section 4 we point out some improvements for these schemes. These improvements can resist

our insider attack. The conclusion is drawn in Section 5.

# 2 Brief Review of Hsu et al.'s Scheme

Let $p$ be a large prime, $q$ a prime divisor of $p-1$, $g$ an element of order $q$ over GF($p$), and $h(\cdot)$ a secure one-way hash function. The parameters $(p, q, g)$ and the function $h()$ are made public. The private key and the public key for each user $p_i$ are $x_i \in Z_q$ and $y_i = g^{x_i} \bmod p$, respectively. Suppose that an original signer $p_0$ with a key pair $(x_0, y_0)$ wants to delegate his/her signing power to a proxy group $PG = \{p_1, p_2, \ldots, p_n\}$ of $n$ proxy signers in such a way that a proxy signature can be created by any subset of $t$ or more proxy signers from PG. Hsu's scheme can be divided into the following two stages: proxy share generation stage and proxy signature generation stage.

## 2.1 Proxy Share Generation

For delegating the signing power to PG, the original signer $p_0$ computes and broadcasts $\widetilde{r} = g^{\widetilde{k}} \bmod p$, where $\widetilde{k} \in_R Z_q$. Once receiving $\widetilde{r}$, each proxy signer $p_i \in PG$ computes and broadcasts $r_i = g^{\alpha_i}\widetilde{r} \bmod p$, where $\alpha_i$ is chosen from $Z_q$ such that $r_i \in Z_p^*$. Upon collecting all $r_i$'s from $p_i \in PG$, $p_0$ computes

$$\widetilde{s} = x_0 h(r, PGID) + n\widetilde{k} \bmod q,$$

where $r = \prod_{i=1}^{n} r_i \bmod p$. Here, $PGID = \{EM, Time, Group\}$ is the proxy group identity which records the proxy status, in which EM denotes the event mark of the proxy share generation including the parameters $t$ and $n$, Time denotes the expiration time of delegation of signing power, and Group denotes the identities of the original signer and the proxy signers of PG. Then $p_0$ performs a $(t, n)$- VSS scheme to share $\widetilde{s}$ among $n$ proxy signers in PG. The share for $p_i$ of $\widetilde{s}$, denoted as $\widetilde{s}_i = f''(i)$, is sent to $p_i$ secretly, where

$$f''(x) = \widetilde{s} + a_1'' x + a_2'' x^2 + \ldots + a_{t-1}'' x^{t-1} \bmod q.$$

For validating the share, $p_0$ publishes $c_i'' = g^{a_i''} \bmod p$ for $i = 1, 2, \ldots, t-1$. Upon receiving $\widetilde{s}_i$ from $p_0$, each $p_i$ can verify it by checking that

$$g^{\widetilde{s}_i} = y_0^{h(r,PGID)} \widetilde{r}^n \prod_{j=1}^{t-1} (c_j'')^{i^j} \bmod p.$$

Note that $\widetilde{s}$ is unknown to all proxy signers. Each proxy signer $p_i$ performs a $(t, n)$-VSS scheme for distributing $f_i(j)$ to proxy signer $p_j$ (for $1 \le j \le n$ and $j \ne i$) via a secure channel, where

$$f_i(x) = \alpha_i + x_i h(r, PGID) + a_{i,1}x + a_{i,2}x^2 + \cdots + a_{i,t-1}x^{t-1} \bmod q.$$

Moreover, $p_i$ broadcasts $c_{i,k} = g^{a_{i,k}} \bmod p$ for $k = 1, 2, \ldots, t-1$. The validity of $f_i(x)$ can be verified by the equality

$$g^{f_j(i)} = r_j \widetilde{r}^{-1} y_j^{h(r,PGID)} \prod_{k=1}^{t-1} (c_{j,k})^{i^k} \bmod p.$$

If all $f_j(i)$'s are verified, each proxy signer $p_i$ computes his/her proxy share as $x_i' = f(i)$, where $f(x) = \sum_{j=1}^{n} f_j(x) \bmod q$. Note that

$$f(0) = \sum_{i=1}^{n} \alpha_i + \sum_{i=1}^{n} x_i h(r, PGID) \bmod q.$$

## 2.2 Proxy Signature Generation

Let $m$ be the message to be signed. Without loss of generality, let $p_1, p_2, \ldots, p_t$ be $t$ proxy signers who want to cooperatively generate a proxy signature. Each participant proxy signer $p_i$ performs a $(t, t)$-VSS scheme by randomly choosing a $(t-1)$-degree polynomial $f_i'(x) = \sum_{j=0}^{t-1} a_{i,j}' x^j \bmod q$ and broadcasts $c_{i,j}' = g^{a_{i,j}'} \bmod p$ for $j = 0, 1, \ldots, t-1$. Then $p_i$ computes $f_i'(j)$ and sends it to $p_j$ via a secure channel for $1 \le j \le n$ and $j \ne i$. Moreover, each participant proxy signer $p_i$ can get

$$x_i'' = f'(i) = \sum_{j=1}^{t} f_j'(i) \bmod q,$$

where $f'(x) = \sum_{j=1}^{t} f_j(x) \bmod q$ and $Y = \prod_{k=1}^{t} c_{k,0}' \bmod p$.

Finally, each $p_i$ computes and broadcasts $T_i = (x_i' + \widetilde{s}_i)h(m) + x_i'' Y \bmod q$.

After validating $T_i$, each $p_i$ computes

$$T = (f(0) + \widetilde{s})h(m) + f'(0)Y \bmod q$$

by applying Lagrange interpolating polynomial. That is

$$T = \sum_{i=1}^{t} T_i \prod_{j=1, j\ne i}^{t} \frac{0-j}{i-j.}$$

Consequently, the proxy signature of $m$ is ($r$, $PGID$, $Y$, $T$). The verification equation of the proxy signature of the message $m$ with respect to the proxy group PG and the original signer is

$$g^T = ((y_0 \prod_{i=1}^{n} y_i)^{h(r,PGID)} r)^{h(m)} Y^Y \bmod p.$$

# 3 Inside Attack on the Hsu et al. Scheme

We suppose that an insider attacker – proxy signer $p_1$ wants to get a threshold proxy signature on message $m'$.

The proxy signer $p_1$ and other participant proxy signer $p_i(2 \le i \le t)$ performs a $(t,t)-$VSS scheme by randomly choosing a $(t-1)$-degree polynomial $f_i'(x) = \sum_{j=1}^{t-1} a_{i,j}' x^j \bmod q$. Then $p_i(2 \le i \le t)$ broadcast $c_{i,j}' = g^{a_{i,j}'} \bmod p$ for $j = 0, 1, \ldots, t-1$. But $p_1$ wait until other participant proxy signer $p_i(2 \le i \le t)$ have broadcast. $p_1$ computes $c_{1,j}' = g^{a_{1,j}'} \bmod p$ for $j = 0, 1, \ldots, t-1$. He privately computes $Y = \prod_{k=1}^{t} c_{k,0}' \bmod p$. Now let $d = h(m')h(m)^{-1} \bmod q$, $Y' = Y^d \bmod p$. He privately computes $c_1' = Y' \cdot (\prod_{k=2}^{t} c_{k,0}')^{-1} \bmod p$. So $p_1$ broadcasts $c_{1,j}'$ for $j = 1, 2, \ldots, t-1$, and broadcasts $c_1'$ instead of $c_{1,0}'$. Each participant proxy signer $p_i$ can get

$$x_i'' = f'(i) = \sum_{j=1}^{t} f_j'(i) \bmod q,$$

where $f'(x) = \sum_{j=1}^{t} f_j'(x) \bmod q$ and obtain $Y' = \prod_{k=2}^{t} c_{k,0}' \cdot c_1' \bmod p$.

Therefore, each $p_i$ computes and broadcasts $T_i' = (x_i' + \widetilde{s}_i)h(m) + x_i'' Y' \bmod q$. Now $p_1$ computes $T' = d \cdot \sum_{i=1}^{t} T_i' \prod_{j=1, j\neq i}^{t} \frac{0-j}{i-j} \bmod q$. Then $(r, PGID, Y', T')$ is a valid threshold proxy signature of message $m'$. Because

$$\begin{aligned} g^{T'} &\equiv g^{d\sum_{i=1}^{t} T_i' \prod_{j=1,j\neq i}^{t} \frac{0-j}{i-j}} \\ &\equiv g^{d\sum_{i=1}^{t}((x_i'+\widetilde{s}_i)h(m)+x_i''Y') \prod_{j=1,j\neq i}^{t} \frac{0-j}{i-j}} \\ &\equiv g^{d((f(0)+\widetilde{s})h(m)+f'(0)Y')} \\ &\equiv ((y_0 \prod_{i=1}^{n} y_i)^{h(r,PGID)} r)^{dh(m)} Y^{dY'} \\ &= ((y_0 \prod_{i=1}^{n} y_i)^{h(r,PGID)} r)^{h(m')} Y'^{Y'} \bmod p. \end{aligned}$$

# 4 Improvements

We now present some effective countermeasures to thwart our attack. First of all, notice that our attack is successful due to the fact that the malicious proxy signer $p_1$ can reveal the value $c_{1,0}'$ after he already knew the values $c_{k,0}'(2 \le k \le t)$'s generated by other honest proxy signers. Therefore, to improve the Hsu et al.'s scheme we can further require that each member should publish his individual value $c_{i,0}'(1 \le i \le t)$ *simultaneously*. Though this seems difficult in the scenarios of computer networks and distributed computing, we can exploit some cryptographic techniques to implement this requirement. One simple way is to require that all members should first commit their $c_{i,0}'$'s and then open their commitments to reveal $c_{i,0}'$'s by using some standard cryptographic commitment schemes. Another method is to require that before generating partial signatures, each member should prove his knowledge of the discrete logarithm of $c_{i,0}'$ to the base $g$ by using interactive or non-interactive knowledge proof protocols [2].

Alternatively, our attack can be avoided by properly modifying the proxy signature generation equation. The simplest way seems to be that adding the value $Y$ into the inputs of the hash function. That is, we now replace all occurrences of $h(m)$ by $h(Y,m)$ in the whole Hsu et al.'s scheme. Therefore, if a dishonest proxy signer wants to forge a proxy signature for another message by mounting the above internal attack, he has to compute a value $Y'$ such that both of the following equations are satisfied:

$$\begin{aligned} d &= h(Y,m)^{-1} \cdot h(Y', m') \bmod p, \\ Y' &= Y^d \bmod p. \end{aligned}$$

However, this is difficult due to the fact that the hash function $h(\cdot)$ is assumed to be a one-way pseudorandom function. Namely, it is infeasible to find a new message $m'$ such that $d = h(Y,m)^{-1} \cdot h(Y^d \bmod p, m', ) \bmod p$ for any number $d$, when the values of $Y, m$ are given. Actually, this improvement is inspired by the famous Schnorr signature scheme [14], where a similar technique is used.

# 5 Conclusions

In this paper, we identify an insider attack on the Hsu et al. threshold proxy signature scheme. That is, a malicious proxy signer can forge a proxy signature for any message. In addition, we pointed out why the original security argument for internal attacks is incorrect, and then proposed some effective improvements to thwart this attack.

# References


[1] A. Bakker, M. Steen, and A. S. Tanenbaum, "A law-abiding peer-to-peer network for free-software distribution," in *IEEE International Symposium on Netwok Computing and Applications (NCA'01)*, pp. 60-67, 2001.

[2] J. Camenisch and M. Stadler, "Efficient group signature schemes for large groups," in *Advances in Cryptology-CRYPTO'97*, LNCS 1294, pp. 410-424, Springer-Verlag, 1997.

[3] I. Foster, C. Kesselman, G. Tsudik, and S. Tuecke, "A security architecture for computational grids," in *Proc. 5th ACM Conference on Computer and Communications Security (CCS'98)*, pp. 83-92, 1998.

[4] C. L. Hsu, T. S. Wu, and T. C. Wu, "Improvement of threshold proxy signature scheme," *Appl. Math. Compu.*, vol. 136, pp. 315-321, 2003.

[5] M. S. Hwang, Eric J. L. Lu, and I. C. Lin, "A practical $(t,n)$ threshold proxy signature scheme based on the RSA cryptosystem," *IEEE Transactions on*

*Knowledge and Data Engineering*, vol. 15. no. 6, pp. 1552-1560, Nov./Dec. 2003.

[6] B. Lee, H. Kim, and K. Kim, "Strong proxy signature and its applications," in *Proceedings of the 2001 Symposium on Cryptography and Information Security (SCIS'01)*, vol. 2/2, pp. 603-608, 2001.

[7] B. Lee, H. Kim, and K. Kim, "Secure mobile agent using strong non-designated proxy signature," in *Information Security and Privacy (ACISP'01)*, LNCS 2119, pp. 474-486, Springer-Verlag, 2001.

[8] J. Leiwo, C. Hanle, P. Homburg, and A. S. Tanenbaum, "Disallowing unauthorized state changes of distributed shared objects," in *Information Security for Global Information Infrastructures (SEC'00)*, pp. 381-390, 2000.

[9] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signatures: delegation of the power to sign messages," *IEICE Transactions on Fundamentals of Electronic Communications and Computer Science*, vol. E79-A, no. 9, pp.1338-1354, 1996.

[10] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signatures for delegating signing operation," in *Proceedings of the 3.th ACM Conference on Computer and Communications Security*, pp. 48-57, 1996.

[11] T. Okamoto, M. Tada, and E. Okamoto, "Extended proxy signatures for smart cards," in *Information Security Workshop (ISW'99)*, LNCS 1729, pp. 247-258, Springer-Verlag, 1999.

[12] H. U. Park and I.Y. Lee, "A digital nominative proxy signature scheme for mobile communications," in *Information and Communications Security (ICICS'01)*, LNCS 2229, pp. 451-455, Springer-Verlag, 2001.

[13] H. M. Sun, N. Y. Lee, and T. Hwang, "Threshold proxy signatures," *IEE Proc., Comput. Digit. Tech.*, vol. 146, no. 5, pp. 259-263, 1999.

[14] C. Schnorr, "Efficient signature generation by smart cards," *Journal of Cryptography*, vol. 4, no. 3, pp. 161-174, 1991.

[15] S. F. Tzeng, M. S. Hwang, C. Y. Yang, "An improvement of nonrepudiable threshold proxy signature scheme with known signers," *Computers & Security*, vol. 23, no. 2, pp. 174-178, 2004.

[16] C. Y. Yang, S. F. Tzeng, and M. S. Hwang, "On the efficiency of nonrepudiable threshold proxy signature scheme with known signers," *Journal of Systems and Software*, vol. 73, no. 3, pp. 507-514, 2004.

[17] K. Zhang, "Threshold proxy signature schemes," in *1997 Information Security Workshop*, pp. 191-197, 1997.

**Lifeng Guo** received B.S. degree in Department of Mathematics from Yanbei Normal University, Shanxi, P.R. China in 2000 and M.S. degree in Department of Mathematics in 2003 from Shanxi University, Shanxi, P.R. China. She is currently pursuing her PhD degree in Institute of Systems Science, Academy of Mathematics and System Sciences, Chinese Academy of Sciences, Beijing, P.R. China. Her current research interests include applied cryptography and computer security.

**Xiangguo Cheng** received his B.S. degree in Mathematics Science from Jilin University in 1992 and his M.S. degree in Applied Mathematics Science from Tongji University in 1998. He is currently a doctoral candidate under the instruction of Prof. Xinmei Wang at the State Key Laboratory of Integrated Services Network of Xidian University, P.R.China. His research interests are in the areas of information theory, Cryptography, and public key cryptosystems.

**Yang Liu** received B.S. degree in Department of Computer Science from Peking University, Beijing, China in 2003. She is currently pursuing master's degree in Computer Science in School of Electronic Engineering and Computer Science in Peking University. For the time being, she is doing some research in P2P security and applied cryptography.