Security Pitfalls of an Efficient Threshold Proxy Signature Scheme for Mobile Agents

Yong Yu\textsuperscript{a,b,*}, Yi Mu\textsuperscript{b}, Willy Susilo\textsuperscript{b}, Man Ho Au\textsuperscript{b}

\textsuperscript{a}School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, 610054, PR China
\textsuperscript{b}School of Computer Science and Software Engineering, University of Wollongong, Wollongong, NSW 2522, Australia

Abstract

A \((t, n)\) threshold proxy signature scheme enables an original signer to delegate his/her signing power to \(n\) proxy signers such that any \(t\) or more proxy signers can sign messages on behalf of the original signer, but \(t - 1\) or less of them cannot produce a valid proxy signature. Based on the RSA cryptosystem, Hong proposed an efficient \((t, n)\) threshold proxy signature for mobile agents. Cai et al. found that the scheme due to Hong is proxy-protected, meaning that the original signer can generate a valid proxy signature by himself. However, it is unclear whether the scheme can be used in reality after fixing the security problem discovered by Cai et al.’s. In this letter, we provide a detailed analysis on Hong’s scheme and show that the scheme fails to achieve the properties of secrecy, proxy protected, undeniability, identifiability and even time constraint and thus adopted of this efficient construction in practice is not recommended.

Key words: Cryptography, Digital signature, Proxy signature, RSA cryptosystem, Security analysis

1 Introduction

The notion of proxy signature [1] was invented by Mambo et al., which enables a proxy signer to sign messages on behalf of an original signer in case of say, temporal absence, lack of computational power etc. After validating the correctness of a proxy signature following a given verification algorithm, a verifier

* Corresponding author.
Email address: yyucd2012@gmail.com (Yong Yu).
can be convinced of the original signer’s agreement on the signed message. In
the last years, fruitful achievements [2–9] of proxy signatures have been seen,
including novel constructions, analysis, improvements and applications. Proxy
signatures have found extensive uses in numerous practical applications such
as in distributed computing, e-commerce, e-cash, and grid computing where
delegation of rights is quite common [2,3]. Mambo et al. [1] classified this kind
of cryptographic primitive into two categories, namely, proxy-unprotected and
proxy-protected. A proxy-protected scheme, where only the proxy signer is
able to generate valid proxy signatures, is more practical since it accommoda-
tes some highly desirable properties such as fairness and signature owner-
ship. In the latest research, delegation with warrant is popular because of its
high security and flexible delegation policy for a proxy signature scheme.

Among all the variants of proxy signature schemes, threshold proxy signature
is one of the useful cryptographic primitives. In a \((t, n)\) threshold proxy signa-
ture scheme, the original signer delegates and distributes the signing power to
\(n\) proxy signers such that collaborative effort of at least \(t\) proxies is required
to the creation of a valid proxy signature, while \(t - 1\) or less of them cannot
complete a signing operation. Threshold proxy signature is a promising prim-
itive for it allows the original signer to control the delegation of his signing
capability. Not only does it allow the original signer to choose the group of
proxies, but also the selection of the threshold value. Thus, to some extent,
threshold proxy signatures are more flexible and practical than traditional
proxy signature schemes. Based on the tricks of secret sharing and threshold
schemes for the first time independently. Subsequently, Sun et al. [12] extended
the construction due to Kim et al. and presented a non-repudiable threshold
proxy signature scheme with known signers. Unfortunately, Hsu et al. [13]
found the scheme in [12] suffers from the conspiracy attack, namely, any \(t\) or
more proxy signers can get the secret keys of other proxy signers. Hwang et
al. [14] proposed a new threshold proxy signature scheme with known signers
and claimed their construction can achieve all the desirable security properti-
es of a proxy signature scheme. However, Wang et al. [15] identified several
security weaknesses in the scheme and concluded that the scheme is not se-
cure. Observing there are few secure \((t, n)\) threshold proxy signature schemes
based on the RSA cryptosystem, Hong [16] proposed a novel and practical
\((t, n)\) threshold proxy signature from RSA mechanism by applying the tradi-
tional RSA cryptosystem without using additional cryptographic techniques,
and suggested to apply the proposal to mobile agent systems. They claimed
that the scheme satisfies all the desirable security requirements. Unfortunately,
Cai et al. [17] demonstrated an concrete attack against Hong’s construction in
which a malicious original signer can forge a valid threshold proxy signature.

**Our contributions:** It is not an easy task to construct a secure threshold
proxy signature scheme from the well-studied RSA problem since sharing the
private key of the RSA system [20] among multiple members is difficult and
the Euler phi function of the modulus cannot be leaked to any proxy signer.
Hong’s scheme [16] has many advantages over other schemes in the same style
such as it shares the proxy signing key using the simple Lagrange formula; the
proxy signature generation and combination are completely non-interactive;
the sizes of both the partial proxy signing key and partial proxy signature are
independent of the number of the proxy signers. It is interesting to find out
whether we can use Hong’s scheme in reality after fixing the security problem
identified by Cai et al. [17]. Unfortunately, in this letter, after giving a detailed
analysis of Hong’s scheme, we find that the construction fails to achieve all the
security properties of a secure proxy signature scheme, including secrecy, proxy
protected, undeniability, identifiability and even time constraint (prevention
of misuse).

Organization: Section 2 reviews Hong’s threshold proxy scheme. Section 3
describes our security analysis of the scheme. Section 4 concludes the paper.

2 Review of Hong’s scheme

The following notations are used in Hong’s scheme [16]. The original signer
is denoted by $U_0$, the $n$ proxy signers are denoted by $U_1, \ldots, U_n$, and a com-
biner is denoted by $C$. $H$ is a secure hash function; $m_w$ denotes a warrant,
which specifies the identities of the original signer and the proxy signers, the
parameters $(t, n)$, the valid delegation period and the kind of messages being
delegated etc. $Q_N$ denotes the subgroup of squares in $\mathbb{Z}_N^*$. The following five
phases are involved in the scheme.

Setup: The original signer $U_0$ picks two large secure primes of equal length
$p_0, q_0$ and computes $N_0 = p_0q_0$, where $p_0 = 2p'_0 + 1, q_0 = 2q'_0 + 1$ with
$p'_0, q'_0$ themselves prime. Let $M_0 = p'_0q'_0$, which is the order of the group $Q_{N_0}$. $U_0$
computes her RSA exponents $e_0$ and $d_0$ such that $e_0d_0 \equiv 1 \pmod{M_0}$. The
private key of $U_0$ is $(d_0, M_0)$ and the public key is $(e_0, N_0)$. Each proxy signer
$U_i (i = 1, 2, \ldots, n)$ chooses two random large secure primes of equal length
$p_i$ and $q_i$, and computes $N_i = p_iq_i, \phi(N_i) = (p_i - 1)(q_i - 1)$, $e_i$ and $d_i$ where
$e_id_i \equiv 1 \pmod{\phi(N_i)}$. The private key and the public key of $U_i$ are $d_i$ and
$(e_i, N_i)$ respectively.

Proxy sharing: $U_0$ firstly generates the threshold proxy signing key $D \equiv
d_0 \cdot H(m_w) \pmod{M_0}$, and shares the signing key among the proxy signer
group as follows.
(1) Set $a_0 = D$ and for $1 \leq i < t$, pick at random $a_i$ from $\{0, \ldots, M_0 - 1\}$,
and define a $t - 1$ degree polynomial

$$f(x) \equiv a_0 + a_1x + \cdots + a_{t-1}x^{t-1} \pmod{M_0}. $$
(2) Compute partial proxy signing key $k_i \equiv f(i) \pmod{M_0}$ for each proxy signer $U_i$.

(3) For the purpose of share validation, $U_0$ picks a random element $v \in Q_{N_0}$ and computes $v_i = v^{k_i}$ for $1 \leq i \leq n$. $U_0$ makes $(v, v_1, \cdots, v_n)$ public and sends $k_i$ to $U_i$ in a secure manner.

**Proxy signature generation:** Assume $t$ different proxy signers $U_i (i = 1, \cdots, t)$ would like to generate a proxy signature of message $m$ on behalf of $U_0$ cooperatively. Let $x = H(m, m_w)$ and $\Delta = n!$. Each proxy signer computes the partial proxy signature $x_i = x^{2\Delta \cdot k_i} \in Q_{N_0}$. Then, $U_i$ computes $\Delta \sigma_i = [x_i / N_i], \sigma_i \equiv x_i^{d_i} \pmod{N_i}$. To guarantee soundness, $U_i$ produces a proof that the discrete log of $x_i^2$ to the base $\hat{x} = x^{4\Delta}$ equals to the discrete log of $v_i$ to base $v$. Specifically, $U_i$ chooses a random $r \in \{0, \cdots, 2^{|N_0|+2L_1} - 1\}$, where $L_1$ is a secondary security parameter, and a secure hash function $H'(\cdot)$, then computes $v' = v^r, \hat{x}' = \hat{x}^r, c_i = H'(v, \hat{x}, v_i, x_i^2, v', \hat{x}')$, $z_i = k_i c + r$. The final partial proxy signature due to $U_i$ is $(i, \Delta \sigma_i, \sigma_i, c_i, z_i)$.

**proxy signature combining:** The combiner $C$ can be one of the proxy signers or a secretary who does not possess any secret parameter. Upon receiving the partial proxy signature $(i, \Delta \sigma_i, \sigma_i, c_i, z_i)$ from $U_i$, $C$ computes $x_i = \Delta \sigma_i \times N_i + (\sigma_i^{c_i} \pmod{N_i})$ and then checks if

$$c = H'(v, \hat{x}, v_i, x_i^2, v^z v_i^{-c}, \hat{x}^z x_i^{-2c}).$$

If the equation holds, the partial proxy signature is valid; Otherwise, invalid.

Assume $t$ partial proxy signatures are valid and without losing generality, the corresponding proxy signers set is $s = \{1, \cdots, t\} \subset \{1, \cdots, n\}$. The proxy signature $w$ of the message $m$ under the warrant $m_w$ is

$$w = x_1^{2\lambda_{0,1}^S} \cdots x_t^{2\lambda_{n,t}^S} \pmod{N_0},$$

where $\lambda_{i,j}^S = \frac{a_i}{\prod_{j' \in S} (i-j')} \prod_{j' \in S} (j-j')$.

Since $w^{e_0} \equiv x^{4\Delta^2 H(m_w)} \pmod{N_0}$ and $\gcd(4\Delta^2, e_0) = 1$, it is easy to find out the final proxy signature $y$ such that $y^{e_0} \equiv x^{H(m_w)} \pmod{N_0}$ by using a standard algorithm $y = w^{a} x^{b}$, where $a, b$ are integers such that $4 \Delta^2 a + e_0 b = 1$.

**Proxy signature verification:** Receiving a $(t, n)$ threshold proxy signature $y$ of the message $m$ under the warrant $m_w$, a verifier checks that if $y^{e_0} \equiv x^{H(m_w)} \pmod{N_0}$ holds, where $x = H(m, m_w)$.

### 3 Security analysis of the scheme

A comprehensive security analysis was provided in [16] and the scheme was claimed to achieve all the desirable security requirements. Unfortunately, in this section, we present several security weaknesses in the scheme and thus show that the claim is not valid.

---

1 This is a typo in [16], and the correct one is $y = w^a x^{bH(m_w)}$
3.1 Secrecy

This property says that even if all the proxy signers collude, they cannot derive the original signer’s private key. In [16], it was claimed that both signing and verification algorithms are based on the RSA problem, one cannot derive the proxy signing key \( D = d_0 \cdot H(m_w) \) (mod \( M_0 \)). Moreover, even if \( t \) out of \( n \) proxy signers conspire to obtain the group proxy signature key \( D \), they cannot derive the original signer’s private key \( d_0 \). Below we demonstrate that the claim does not hold.

Firstly, for any set \( \Omega \subseteq \{1, 2, \ldots, n\} \) satisfying \( |\Omega| \geq t \), we can get \( \sum \lambda_{\Omega,j} \cdot f(j) \) (mod \( M_0 \)) using the Lagrange formula. Thus, any \( t \) or more colluding proxy signers are able to compute the proxy signing key \( D \). Worse still, since \( D \equiv d_0 \cdot H(m_w) \) (mod \( M_0 \)), we can get

\[
e_0 \cdot D \equiv e_0 \cdot d_0 \cdot H(m_w) \equiv H(m_w) \mod M_0,
\]

which means that \( e_0 \cdot D - H(m_w) \) is a nonzero multiple of \( M_0 \), where \( M_0 = p_0q_0 = \phi(N_0) \). However, following the well-known conclusion due to Miller [18], knowing such a multiple of \( \phi(N_0) \) is equivalent to factoring \( N_0 \). Consequently, the \( t \) colluding proxy signers, after obtaining the value \( D \), is capable of factoring \( N_0 \) and thus computing the secret key of the original signer. Specifically, with the factors of \( N_0 \), the colluding proxy signers can compute the Euler’s phi function \( \phi(N_0) \), and they can get the private key \( d_0 \) from \( e_0d_0 \equiv 1 \) (mod \( M_0 \)) by employing extended Euclidean algorithm [19]. Therefore, the property of secrecy is violated in the scheme.

3.2 Proxy protected and identifiability

Proxy protection states that only at least \( t \) out of \( n \) proxy signers can cooperate to generate valid proxy signatures. Any third party, including the original signer, cannot masquerade as proxy signers group to generate a valid proxy signature. As identified by Cai et al. [17], since the final proxy signature is \( y \) and the verification equation is \( y^{e_0} \equiv H(m, m_w)^{H(m_w)} \) (mod \( N_0 \)), nobody can recognize the identities of the \( t \) actual proxy signers upon receiving the final proxy signature. That is to say, the property of identifiability cannot be achieved in the scheme [16]. Moreover, they showed that an malicious original signer can forge a valid proxy signature in polynomial time. Thus, the scheme fails to provide the property of proxy protection.
3.3 Unforgeability

According to the analysis above, we can conclude that the scheme is not secure against the attacks due to the malicious original signer and malicious proxy signers, namely, it is insecure against the insider attacks. Here we also demonstrate that it is even possible for an outsider adversary to mount an universal forgery.

(1) Choose an arbitrary warrant $m_w^*$ which records the delegation policy including limits of authority, valid periods of delegation and proxy signature, the threshold value $t$ and the identities and the public keys of the original signer and the proxy signers group.

(2) Compute $H(m_w^*)$ and checks if $e_0 | H(m_w^*)$ holds. If it does not hold, repeats (1) and (2).

(3) Choose a message $m^*$ that conforms to the the warrant $m_w^*$, and compute $y^* \equiv H(m^*, m_w^*)^{k} \pmod{N_0}$ where $H(m_w^*) = k e_0$.

(4) Publish $y^*$ as the threshold proxy signature of message $m^*$ under the warrant $m_w^*$.

It’s easy to check the correctness of the forged proxy signature. Next we assess the success probability of the adversary. The RSA parameters should be thoroughly evaluated for security and efficiency [21, 22] in some applications. To reduce the time of signature verification or encryption in RSA, one may wish to use a small $e$. The smallest value for $e$ is 3, but the value $e = 2^{16} + 1$ is recommended to resist certain attacks [23]. Assume the recommended RSA exponent $e = 2^{16} + 1$ and the standard SHA1 hash algorithm, whose output is of 160-bit, are employed. For a random $m_w^*$, the probability that $e_0 | H(m_w^*)$ is at least $2^{-16}$, which is not negligible. Thus, the expected time for the attacker to find a warrant $m_w^*$ such that $e_0 | H(m_w^*)$ is within evaluating $2^{16}$ hash values on different warrants $m_w^*$. This is highly feasible since modern computer can handle hashing of a file over a 100 MB in less than 1 second\(^2\) and the warrant $m_w^*$ is supposed to be much shorter.

3.4 Undeniability

From the analysis of unforgeability, we can conclude that the scheme can resist neither insider attacks nor outsider attacks. As a consequence, given a threshold proxy signature, both the original signer or the proxy signer group can deny having signed the message.

\(^2\) http://www.cryptopp.com/benchmarks.html.
3.5 Time constraint

According to the universal forgery attack described above, an outsider adversary can arbitrarily modify the warrant \( m^*_w \), including changing the valid periods of delegation, replacing the identities of the original signer and proxy signers, modifying and scope of singing message, altering the threshold value \( t \), and then forge threshold proxy signature as long as \( e_0 | H(m^*_w) \) holds. Thus, the scheme fails to achieve the property of prevention of misuse as well.

4 Conclusion

In this letter, we presented a detailed security analysis of the threshold proxy signature proposed in [16], and showed that it fails to achieve the properties of secrecy, proxy protected, undeniability, identifiability as well as prevention of misuse. Thus, its adoption in practice is not recommended. Hopefully the analysis is helpful for the future constructions of threshold proxy signature from RSA assumption.

Acknowledgement: The first author is supported by the University of Wollongong Vice Chancellor Fellowship. The third author is supported by Australian Research Council Future Fellowship FT0991397. This work is supported by the NSFC of China under Grant 61003232, 61103207, U1233108, 61272436, the National Research Foundation for the Doctoral Program of Higher Education of China under Grant 20100185120012, the NSFC of China for International Young Scientists under Grant 61250110543, and the Fundamental Research Funds for the Central Universities under Grants ZYGX2010J066 and ZYGX2011J067.

References


A. Boldyreva, A. Palacio, B. Warinschi, Secure proxy signature scheme for

Z. Shao, Proxy signature schemes based on factoring, Information Processing

Y. Sun, C. Xu, Y. Yu, Y Mu, Strongly unforgeable proxy signature scheme
secure in the standard model, Journal of Systems and Software 84(9) (2011)
1471-1479.

H. Huang, Y. Mu, W. Susilo, F. Zhang, X. Chen, A short proxy signature
scheme: efficient authentication in the ubiquitous world, in: Proceeding of
2th International Symposium on Ubiquitous Intelligence and Smart Worlds,
Nagasaki, Japan, 2005, pp. 480-489.

F. Zhang, R. S. Naini, C. Lin, Some new proxy signature schemes from bilinear
pairings, in: Progress on Cryptography: 25 years of Cryptography in China,
Kluwer International Series in Engineering and Computer Science, vol. 769,

Y. Yu, Y. Mu, W. Susilo, Y. Sun, Y. Ji, Provably secure proxy signature scheme
from factorization, Mathematical and Computer Modelling, 55 (3-4) (2012)
1160-1168.

K. Zhang, Threshold proxy signature schemes, in: Proceeding of the Information

S. Kim, S. Park, D. Won, Proxy signature, revisited, in Proceeding of ICICS 1997,

H. Sun, N. Lee, T. Hwang, Threshold proxy signature, in: Proceeding of

C. Hsu, T. Wu, D. Wong, New nonrepudiable threshold proxy signature scheme
with known known signers, Journal of System and Software, 58 (2) (2001) 119-
124.

M. Hwang, J. Lu, L. Lin, A practical (t, n) threshold proxy signature scheme
based on the RSA cryptosystem, IEEE Transcations on Knowledge and data

G. Wang, F. Bao, J. Zhou, R. Deng, Comments on a threshold proxy signature
scheme based on the RSA cryptosystem, IEEE Transcations on Knowledge and

X. Hong, Efficient threshold proxy signature protocol for mobile agents,

X. Cai, S. Wang, Cryptanalysis of efficient threshold proxy signature protocol
for mobile agents, in: Proceeding of 2011 IEEE International Conference on
726-728.


