Further Cryptanalysis of some Proxy Signature Schemes

Jiqiang Lv, Jingwei Liu and Xinmei Wang

National Key Lab of Integrated Service Networks(ISN)
Xi'an Electronics Science&Technology University(XIDIAN),
Taibai Road, Xi'an City, Shaanxi Province, 710071 CHINA
lvjiqiang@hotmail.com,jwliu@mail.xidian.edu.cn,
xmwang@xidian.edu.cn

Abstract. Proxy signature is a signature that an original signer delegates his or her signing capability to a proxy signer, and then the proxy signer creates a signature on behalf of the original signer. However, Sun *et al.* [7] showed that the proxy and multi-proxy signatures of Lee *et al.*[3], and the strong proxy signature scheme with proxy signer privacy protection of Shum *et al.*[6] are not against the original signer's forgery attack, so these schemes do not process the property of unforgeability. In this paper, we present an extensive forgery method on these schemes, which makes the forgery method of H.Sun *et al.* be a special case of ours.

Key words: Public Key Cryptography; Proxy Digital Signature; Forgery Attack

1 Introduction

Proxy signature is a signature that an original signer delegates his or her signing capability to a proxy signer, and then the proxy signer creates a signature on behalf of the original signer. Mambo *et al.*[5] firstly gave a systematic discussion of proxy signatures and classified proxy signatures based on delegation type as full delegation, partial delegation and delegation by warrant. Under the full delegation, the original signer gives his secret key to the proxy signer. Under the partial delegation, the original signer generates a proxy signature key by using his secret key and gives it to the proxy signer. The proxy signer uses the proxy key to sign. Accordingly, the verification equation for proxy signature is modified, so that the proxy signature is distinguishable from the signature signed by the original signer. Under the delegation by warrant, the proxy signer obtains the warrant which is a certificate composed of a message part and a public signature key from the original signer and uses the secret key to sign. The resulting signature consists of the created signature and the warrant. Later, Zhang [8] proposed two threshold proxy signature with the property of non-repudiaibility which means that both original signer and proxy signer cannot repudiate his participation in a signature. Lee *et al.*[3] provided new classifications of proxy signatures as strong vs. weak proxy signatures, designated vs. non-designated proxy signatures and self-proxy signatures. Strong proxy signature represents both original signer's and proxy signer's signatures. Once a proxy signer creates a valid proxy signature, he cannot repudiate his signature creation against anyone. Weak proxy signature represents only original signer's signatures. It does not provide the non-repudiation of proxy signer. Shum *et al.*[6] proposed a proxy signer-protected signature, during which, the real identity of a proxy signature is hidden to an alias. Only under the help of an authority could the real identity be revoked. However, Sun *et al.*[7] showed that some exiting proxy signature are not against the original signer's forgery attack, so the schemes do not process the unforgeability property.

In this paper, we present an extensive forgery attack on the proxy and multi-proxy signatures of Lee *et al.*, and the strong proxy signature scheme with proxy signer privacy protection of Shum *et al.* During the forgery attack on these schemes, a dishonest original signer can generate his users' secret so that he could cheat under the name of his users without detecting.

The rest of the paper is organized as follows. In the next section, we list some notations and domain parameters. In Section 3, we show our extensive forgery attack on the proxy and multi-proxy signatures of Lee *et al.* In Section 4, we show our extensive forgery attack on the strong proxy signature scheme with proxy signer privacy protection of Shum *et al.* A simple conclusion is made in the final section.

2 Notation and Domain Parameters

Throughout this paper, we will use the following notations and parameters.

p: a public large prime

q: a public large prime factor of $p-1$

g: a public base element of order q in Z_p

$h(.)$: a public one-way hush function

x_a: original signer *Alice*'s secret key with the corresponding public key $y_a = g^{x_a} \bmod p$

x_b: proxy signer *Bob*'s secret key with the corresponding public key $y_b = g^{x_b} \bmod p$

m_ω: a warrant

3 Forgery Attack on the Proxy and Multi-Proxy Signature Schemes of Lee *et al.*

3.1 Lee *et al.*'s Proxy Signature Scheme

Proxy Key Generation

The original signer *Alice* randomly selects $k_0 \in Z_q^*$, and computes $r_0 = g^{k_0} \bmod p$ and $s_0 = x_a \cdot h(m_\omega, r_0) + k_0 \bmod p$. Then she sends (m_ω, r_0, s_0) to the proxy signer *Bob*.

Bob accepts (m_ω, r_0, s_0) as a valid proxy key from *Alice* if $g^{s_0} = {y_a}^{h(m_\omega, r_0)} \cdot r_0$ holds. Then, *Bob* computes his proxy key pair (x_p, y_p) as $x_p = x_b + s_0 \bmod q, y_p = g^{x_p} = y_b \cdot r_0 \cdot y_a^{h(m_\omega, r_0)} \bmod p$.

Proxy Signature Generation

In order to create a proxy signature on a message M conforming to the warrant information m_ω, *Bob* uses his proxy key pair (x_p, y_p) and a DLP-like signature scheme to generate a signature σ. And he sends $(\sigma, M, r_0, m_\omega)$ to the recipient V.

Signature Verification

V computes the proxy public key $y_p = {y_a}^{h(m_\omega, r_0)} \cdot r_0 \cdot y_b \bmod p$ and uses it to verify the validity of the proxy signature σ by using the DLP-like signature scheme.

3.2 Lee *et al.*'s Multi-Proxy Signature Scheme

Let O_i be the group of n original signers, (x_i, y_i) be their corresponding secret and public key pairs.

Multi-Proxy key generation

The original signer O_i selects $k_i \in Z_q^*$, and computes $r_i = g^{k_i} \bmod p$ and $s_i = x_i \cdot h(m_{\omega_i}, r_i) + k_i \bmod p$. Then he sends (m_{ω_i}, r_i, s_i) to the proxy signer *Bob. Bob* accepts (m_{ω_i}, r_i, s_i) as a valid proxy key from O_i if $g^{s_i} = {y_i}^{h(m_{\omega_i}, r_i)} \cdot r_i \bmod p$ holds.

Proxy signature generation

Bob computes the proxy signature key $x_p = \sum_{i=1}^{n} s_i + x_b$, and generates the proxy signature σ of a message M by using the proxy signature key. Then, he sends $(M, \sigma, m_{\omega_1}, r_1, \ldots, m_{\omega_n}, r_n)$ to the verifier V.

Signature Verification

V computes the proxy public key $y_p = \prod_{i=1}^{n} ({y_i}^{h(m_{\omega_i}, r_i)} \cdot r_i) \cdot y_b \bmod p$, and then verifies the signature σ by using the DLP-like signature scheme.

3.3 Attack on the Proxy Signature Scheme of Lee *et al.*

A dishonest original signer *Alice* randomly selects $a, b \in Z_q^*$, and computes

$$r_0^* = y_b^{-1} \cdot y_a^a \cdot g^b \bmod p,$$

$$x_p^* = x_a \cdot (h(m_\omega, r_0^*) + a) + b \bmod q.$$

Now, *Alice* can impersonate *Bob* by using (r_0^*, x_p^*), since

$$\begin{aligned} & y_a^{h(m_\omega, r_0^*)} \cdot r_0^* \cdot y_b \bmod p \\ = & y_a^{h(m_\omega, r_0^*)} \cdot y_b^{-1} \cdot y_a^a \cdot g^b \cdot y_b \bmod p \\ = & y_a^{h(m_\omega, r_0^*)} \cdot y_a^a \cdot g^b \bmod p \\ = & g^{x_p^*} \bmod p. \end{aligned}$$

Note that by using these invalid proxy keys, a dishonest original signer can produce a valid proxy signature under the name of his proxy signer, so the proxy scheme of Lee *et al.* does not meet the requirement of strong unforgeability.

3.4 Attack on the Multi-Proxy Signature Scheme of Lee *et al.*

In the multi-proxy signature scheme of Lee *et al.*, in order to forge a multi-proxy signature, a dishonest original signer O_j randomly selects $a, b \in Z_q^*$, and computes

$$r_j^* = (y_b \cdot \prod_{i=1, i\neq j}^{n} y_i{}^{h(m_{\omega_i}, r_i)} \cdot r_i)^{-1} \cdot y_j^a \cdot g^b \bmod p,$$

$$x_p^* = x_j \cdot (h(m_{\omega_j}, r_j^*) + a) + b \bmod q.$$

Now, the dishonest original signer O_j can impersonate his proxy signers by using (r_j^*, x_p^*), this is because

$$\begin{aligned}
&y_b \cdot \prod_{i=1}^{n} (y_i{}^{h(m_{\omega_i}, r_i)} \cdot r_i) \bmod p \\
&= y_b \cdot y_j^{h(m_{\omega_j}, r_j^*)} \cdot r_j^* \cdot \prod_{i=1, i\neq j}^{n} (y_i{}^{h(m_{\omega_i}, r_i)} \cdot r_i) \bmod p \\
&= y_b \cdot y_j^{h(m_{\omega_j}, r_j^*)} \cdot (y_b \cdot \prod_{i=1, i\neq j}^{n} y_i{}^{h(m_{\omega_i}, r_i)} \cdot r_i)^{-1} \cdot y_j^a \cdot g^b \cdot \prod_{i=1, i\neq j}^{n} (y_i{}^{h(m_{\omega_i}, r_i)} \cdot r_i) \bmod p \\
&= y_j^{h(m_\omega, r_j^*)} \cdot y_j^a \cdot g^b \bmod p \\
&= g^{x_p^*} \bmod p.
\end{aligned}$$

So the multi-proxy signature scheme of Lee *et al.* does not satisfy the requirement of strong unforgeability, too.

4 Forgery Attack on the Strong Proxy Signature Scheme with Proxy Signer Privacy Protection of Shum *et al.*

4.1 Shum *et al.*'s Proxy Signature Scheme

Based on Lee *et al.*'s proxy signature, Shum *et al.* proposed a privacy-protected proxy signature scheme which protects the proxy signer's identity behind an alias. Their scheme consists of the following five phases: the alias issuing, the proxy delegation, proxy signature generation, signature verification, and the privacy revoking.

Alias Issuing

The proxy signer *Bob* sends his identity ID_b to the alias issuing authority T. T randomly selects two integers $k_p, k_T \in Z_q^*$, and computes $h_p = h(k_p, ID_b)$, $r_T = g^{k_T} \bmod p$ and $s_T = x_T \cdot h(h_p, r_T) + k_T \bmod p$, where x_T is T's secret key. Then T sends (h_p, r_T, s_T) to the proxy signer *Bob*.

Bob accepts (h_p, r_T, s_T) if $g^{s_T} = y_T{}^{h(h_p, r_T)} \cdot r_T \bmod p$ holds.

Proxy Delegation

The original signer *Alice* randomly selects $k_0 \in Z_q^*$, and computes $r_0 = g^{k_0} \bmod p$ and $s_0 = x_a \cdot h(m_\omega, r_0) + k_0 \bmod p$. Then she sends (m_ω, r_0, s_0) to the proxy signer *Bob*.

Bob accepts (m_ω, r_0, s_0) as a valid proxy key from *Alice* if $g^{s_0} = {y_a}^{h(m_\omega, r_0)} \cdot r_0$ holds.

Proxy Signature Generation

Bob computes his proxy secret key $x_p = s_T + s_0 \bmod q$ and corresponding public key $y_p = g^{x_p} \bmod p$, and generates a signature σ for the message M by using a DLP-like signature scheme. Finally he sends $(\sigma, M, r_0, m_\omega, h_p, r_T)$ to the recipient V.

Signature Verification

V computes the proxy public key $y_p = {y_a}^{h(m_\omega, r_0)} \cdot r_0 \cdot {y_T}^{(h_p, r_T)} \cdot r_T \bmod p$ and then verifies the validity of the proxy signature σ by using the DLP-like signature scheme.

Privacy Revoking

V sends the alias h_p to T. Next, T returns k_p and ID_b to V. V accepts that the proxy signer's identity is ID_b if $h_p = h(k_p, ID_b)$ holds.

4.2 Attack on the Proxy Signature Scheme of Lee *et al.*

A dishonest original signer *Alice* randomly selects $a, b \in Z_q^*$, and computes

$$r_0^* = ({y_T}^{(h_p, r_T)} \cdot r_T)^{-1} \cdot y_a^a \cdot g^b \bmod p,$$

$$x_p^* = x_a \cdot (h(m_\omega, r_0^*) + a) + b \bmod q.$$

Now, *Alice* can impersonate *Bob* by using (r_0^*, x_p^*), since

$$\begin{aligned} & y_a^{h(m_\omega, r_0^*)} \cdot r_0^* \cdot {y_T}^{(h_p, r_T)} \cdot r_T \bmod p \\ = & y_a^{h(m_\omega, r_0^*)} \cdot y_a^a \cdot g^b \bmod p \\ = & g^{x_p^*} \bmod p. \end{aligned}$$

Note that a dishonest T can also impersonate his user in the same way. Furthermore, the proxy signature scheme of K.Shum et al. is unpratical, for a verifier can determine by himself whether a signature comes from a person which is once exposed to him by T before.

5 Conclusion

we further cryptanalyze some proxy signature schemes, and present an extensive forgery method, during which a dishonest original signer can impersonate his proxy signers, so the related proxy signature schemes is insecure. Anyway, we should point out that our forgery attack maybe has little value in reality, for a proxy signer can prove the dishonesty of the original signer by showing that he has two different groups of proxy parameters which could only be generated by the original signer, if the recipient sends the invalid original signer-producing proxy signature to him. But in the point of research, this forgery attack can make these proxy signatures insecure.

Acknowledgement: This work was supported by China National 973 Project Found(G1999035804)

References

1. T.ElGamal.: A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms, IEEE Transactions on Information Security,IT-31, pp.469-472,1985.
2. S.Kim, S.Park and D.Won.: Proxy Signatures, revisited, Proc. of ICICS'97-First International Conference on Information and Communication Security, Springer-Verlag, LNCS 1334,pp.223-232, 1997.
3. B.Lee, H.Kim and K.Kim.: Strong Proxy Signature and its Applications, Proc. of SCIS-Sypsium on Cryptology and Information Security, pp.603-608,2001.
4. B.Lee, H.Kim and K.Kim.: Secure Mobile Agent Using Strong Non-designated Proxy Signature, Proc. of ACISP2001, Springer-Verlag, LNCS 2119, pp.474-486, 2001.
5. M.Mambo, K.Usuda and E.Okamoto.: Proxy Signature:Delegation of the Power to Sign Messages, IEICE Trans.Fundamentals,Vol.E79-A:No.9,pp. 1338-1353, 1996.
6. K.Shum and V.Wei.: A Strong Proxy Signature Scheme with Proxy Signer Privacy Protection. Proc. of WETICE'02-Eleventh IEEE International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, pp.55-56, 2002.
7. H.Sun and B.Hsieh.: On the Security of some Proxy Signature Schemes. Cryptology Eprint Achives, Report 2003/068.
8. K.Zhang.: Threshold Proxy Signature Schemes, Proc.of 1st International Information Security Worshop, pp. 282-290,1997.