2002 International Computer Symposium (ICS2002)

(1) Name of the workshop：Workshop on Cryptology and Information Security

(2) Title of the paper：A Proxy-Protected Proxy Multi-Signature Scheme Based on the Elliptic Curve Cryptosystem

(3) A short abstract：The research in the paper contributes to publicly delivering the delegation parameter and to reducing the amount of verifying operation for a proxy signature. A new proxy-protected proxy multi-signature scheme is presented based on the elliptic curve discrete logarithm problem (ECDLP). To the demand for security, the proposed scheme inherits most merits of the typical solutions based on the discrete logarithm problem (DLP). As to the expectation toward efficiency, the scheme on the elliptic curve cryptosystem (ECC) can achieve the performance of the cryptosystem more efficient than those on the DLP.

(4) Name：Tzer-Shyong Chen, Tzuoh-Pyng Liu, and Yu-Fang Chung

Current affiliation：Department of Computer Science and Information Engineering, Da-Yeh University

Postal address：Department of Computer Science and Information Engineering, Da-Yeh University, 112 Shan-Jiau Rd, Da-Tsuen, Changhua, Taiwan 515, R.O.C.

E-mail address：r9006013@mail.dyu.edu.tw

Telephone number：0923696355

(5) Name of the contact author：Tzer-Shyong Chen

(6) A list of keywords：Elliptic Curve Cryptosystem, Elliptic Curve Discrete Logarithm Problem, Proxy Signature, Proxy Multi-Signature and Cryptography

A Proxy-Protected Proxy Multi-Signature Scheme Based on the Elliptic Curve Cryptosystem

Tzer-Shyong Chen Tzuoh-Pyng Liu* Yu-Fang Chung*

Department of Information management, Tunghai University, Taichung, Taiwan 40744, R.O.C.

* Graduate School of Computer Science and Information Engineering, Da-Yeh University

E-mail: arden@mail.dyu.edu.tw

Abstract

The research in the paper contributes to publicly delivering the delegation parameter and to reducing the amount of verifying operation for a proxy signature. A new proxy-protected proxy multi-signature scheme is presented based on the elliptic curve discrete logarithm problem (ECDLP). To the demand for security, the proposed scheme inherits most merits of the typical solutions based on the discrete logarithm problem (DLP). As to the expectation toward efficiency, the scheme on the elliptic curve cryptosystem (ECC) can achieve the performance of the cryptosystem more efficient than those on the DLP.

Key words: Elliptic Curve Cryptosystem, Elliptic Curve Discrete Logarithm Problem, Proxy Signature, Proxy Multi-Signature and Cryptography

1. Introduction

A digital signature is generally applied to the various electronic documents in the digital times. To be provided with both validity and undeniability, a digital signature must be affixed via the secret key held by the signer so that the verifier can determine the validity of signature via the public key equally attached to the same one. It is a common situation that a document cannot become effective except under the proviso of a certain signer who may be not able to sign by himself. Then, the signer can empower a proxy signer to generate a valid signature defined as a proxy signature for him. The proxy signature scheme was first introduced by Mambo *et al*. [1] in 1996. By such a technique, an original signer only can delegate one proxy signer to sign the messages for himself. Later, another securer version [2] was presented by Mambo *et al.*, in which no one can forge the proxy signature even to the original one. Such a property is indicated as "non-repudiated" or " proxy-protected". Different from the one-to-one scheme by Mambo, the concept of proxy multi-signature presented by Yi [3] allows two or more original signers delegate the same proxy signer to sign the messages for all original ones.

According to the authorized degree, the shapes of proxy signatures are differentiated into the following three: *full delegations*, *partial delegations*, and *delegations by warrant*. Kim *et al.* [4] originated to combine both partial delegation and delegation by warrant in 1997, so that the generation of signature by the original or the proxy signers becomes to be identified and the delegation qualification can be limited by the original signer. So far the technique of proxy signature is developed under the considerations of the practical application and requirement, the last originated to combine both partial delegation and delegation by warrant is most in match with the current demands. Therefore, the proposed scheme is directed at such a kind of authorized degree.

As what was mentioned in the Sun′s research [5,6], the public key substitution attack universally occurs in the existing proxy signature schemes. Aimed at the attack, he presented several modified proxy signature schemes to give the solutions, such as the schemes by Mambo, that by Yi, and that by Kim. However, there left something to be improved that the delivery of the delegation parameter needs to be extra-enciphered and extra-deciphered in the Sun′s schemes. Actually, such a kind of enciphering and deciphering processes should be negligible because it will burden the system with overhead. In light of the above-mentioned, the scheme is presented to avoid the scheme from the public key substitution attack under the condition of no extra-overhead for the efficiency of performance.

After being proposed by both Koblitz [7] and Miller [9] in 1985, the elliptic curve has widely applied to the cryptosystems. The security of ECC rests on the difficulty of the ECDLP [7-11]. The ECC is constructed by the integer points over the elliptic curve in the finite fields. The basic operations contain the addition and multiplication operations under the ECC, thus the operations by ECC are more efficient than the other cryptosystems, such as the RSA and DSA. Concerning for performance efficiency and security, the ECC is directed to solving the secure defense problem of a cryptosystem.

In the later sections of the paper, Section 2 illustrates the new proxy multi-signature scheme, and Section 3 emphasizes on the analyses of security and efficiency. Finally, Section 4 concludes the research in various points.

2. The elliptic curve proxy-protected proxy multi-signature scheme

To successfully withstand the public key substitution attack and achieve the delivery of the delegation parameter without the additional enciphering and deciphering procedure, the new one on the ECDLP is presented, which is equally resulted from the proxy-protected proxy multi-signature scheme by Sun on the DLP [6]. Moreover, another improvement is that the proposed multi-signature scheme makes the computation overhead independent from the number of the original signer. The structure of the proposed scheme is divided into four phases, including the system initialization phase, the key generation phase, the proxy signature generation phase, the proxy signature verification phase.

2.1 System initialization phase

Before initializing the whole scheme, the following parameters over the elliptic curve domain are required:

Step 1: A field size p, which is a large odd prime.

Step 2: Two parameters a, $b \in F_p$ to define the equation of elliptic curve E over F_p (i.e., $y^2 = x^3 + ax + b \pmod{p}$ in the case $p > 3$), where $4a^3 + 27b^2 \neq 0 \pmod{p}$. The cardinality of E should be divisible by a large prime number for the security issue of Pohlig and Hellman [10].

Step 3: A finite point $B = (x_B, y_B)$ whose order is a large prime number in $E(F_p)$, where $B \neq O$, because O denotes an infinity point.

Step 4: The order of $B = t$.

2.2 Key generation phase

This phase can be further divided into two parts.

Part 1: Personal public key generation phase

All original signers and the designated proxy signer are authorized to select the secret key owned by the individual.

- For each $1 \leq i \leq n$, the original signer A_i randomly selects a number $d_i \in [1,t\text{-}1]$ in secure, and then computes $Q_i = d_i \times B = (x_{Q_i}, y_{Q_i})$. If $x_{Q_i} \neq 0$, then indicate d_i as the secret key and Q_i as the public one.
- The proxy signer randomly selects a number $d_p \in [1,t\text{-}1]$ in secure, and then computes $Q_p = d_p \times B = (x_{Q_p}, y_{Q_p})$. If $x_{Q_p} \neq 0$, then indicate d_p as the

secret key and Q_p as the public one.

All public keys $\{Q_i\}$ and Q_p must be certified through the signification of the CA, in which $i = 1, 2, \ldots, n$.

Part 2: Proxy-signature secret key generation phase

Step 1: (Secret key generation) For each $1 \leq i \leq n$, the original signer A_i selects a random number $k_i \in \{1, 2, \ldots, t\text{-}1\}\backslash d_i$ in secure as the secret key.

Step 2: (Group commitment value generation) For each $1 \leq i \leq n$, the original signer A_i respectively computes $R_i = k_i \times B = (x_{R_i}, y_{R_i})$, if $x_{R_i} = 0$, then go to step 1, otherwise, broadcast the resulting R_i to the other members. After receiving these available $\{R_i\}$ from the others through the broadcast channel, every member can compute the point $R = \sum_{i=1}^{n} R_i = (x_R, y_R)$, in which the parameter x_R is indicated as a group commitment value.

Step 3: (Sub-delegation parameter generation) For each $1 \leq i \leq n$, the original signer A_i uses his own secret keys d_i, k_i and the group commitment value x_R to compute:

$$s_i = d_i \cdot h(M_w, x_{Q_i}, x_{Q_p}, x_R) - k_i \pmod t$$

Where $h(\ )$ is a public collision resistant hash function and the warrant M_w contains few information, such as the IDs of all original signers, the ID of the proxy signer, and the delegation period, etc. Then, the sub-delegation parameter for A_i is (M_w, s_i).

Step 4: (Sub-delegation parameter delivery) For each $1 \le i \le n$, the original signer A_i sends the sub-delegation parameter (M_w, s_i) to the proxy signer in a public channel.

Step 5: (Sub-delegation parameter verification) Once the proxy signer receives the sub-delegation parameters (M_w, s_i), and then he uses these (M_w, s_i) to compute the following $R_i^{'} = (x_{R_i^{'}}, y_{R_i^{'}})$:

$$R_i^{'} = h(M_w, x_{Q_i}, x_{Q_p}, x_R) \times Q_i - s_i \times B$$

If $x_{R_i^{'}} = x_{R_i}$ (mod t), then he accepts (M_w, s_i) as a valid sub-delegation parameter; otherwise, he rejects it and requests for a valid one toward the certain A_i, or terminates this protocol.

Step 6: (Proxy multi-signature secret key generation) If the proxy signer confirms the validity of all sub-delegation parameters (M_w, s_i) in which $1 \le i \le n$, and then he computes the proxy multi-signature secret key as follows:

$$\bar{d}_p = d_p + \sum_{i=1}^{n} s_i \pmod{t}$$

2.3 Proxy signature generation phase

While signing a message m for A_1, A_2, …, A_n, the proxy signer executes the signing operation aimed at the ordinary signature scheme using the proxy multi-signature secret key $\bar{d}_p$. Assume that the resulting signature is $Sign_{\bar{d}_p}(m)$. The proxy multi-signature on m for A_1, A_2, …, A_n is $(m, Sign_{\bar{d}_p}(m), R, M_w)$. Then, the proxy signer sends the $(m, Sign_{\bar{d}_p}(m), R, M_w)$ to the verifier.

2.4 Proxy Signature Verification Phase

The verifier computes the corresponding proxy multi-signature public key using the ordinary signature scheme:

$$\overline{Q}_p = Q_p + h(M_w, x_{Q_1}, x_{Q_p}, x_R) \times Q_1 + \cdots + h(M_w, x_{Q_n}, x_{Q_p}, x_R) \times Q_n - R$$

In the ordinary signature scheme with the new generated proxy multi-signature public key $\overline{Q}_p$, the verifier confirms the validity of $Sign_{\bar{d}_p}(m)$ by verifying the accuracy of the verification equation.

Theorem 2.1

For each $1 \le i \le n$, if $x_{R_i^{'}} = x_{R_i}$ (mod t), then the proxy signer authenticates the (M_w, s_i) as a valid sub-delegation parameter.

Proof

$$s_i = d_i \cdot h(M_w, x_{Q_i}, x_{Q_p}, x_R) - k_i \pmod t$$

$$\Leftrightarrow k_i = d_i \cdot h(M_w, x_{Q_i}, x_{Q_p}, x_R) - s_i \pmod t$$

$$\Leftrightarrow k_i \times B = [d_i \cdot h(M_w, x_{Q_i}, x_{Q_p}, x_R) - s_i \pmod t] \times B$$

$$\Leftrightarrow k_i \times B = [d_i \cdot h(M_w, x_{Q_i}, x_{Q_p}, x_R) \pmod t] \times B - s_i \times B$$

$$\Leftrightarrow R_i = h(M_w, x_{Q_i}, x_{Q_p}, x_R) \times Q_i - s_i \times B$$

$$\Leftrightarrow R_i^{'}$$

3. Security and Performance Analyses

3.1 Security Issues

Issue 1: ECDLP

The difficulty resulted from ECDLP is based on the derivation of *d* according to the given *B* and *Q* as follows:

$$Q = d \times B$$

In the equation, $d \times B$ indicates that the point B is added to itself for d times and Q is a point derived from $d \times B$, in which Q depends on the number of d. Therefore, an attacker in the proposed scheme encounters the difficulty constituted by the ECDLP, which makes him failed in deriving the private key from the public one to forge the signature.

Issue 2: Public key substitution attack

The signature verification equation is integrated with a one-way hash function and the operation by the ECC. The difficulty, for any attackers to forge another public key from the above equation, is equivalent to the solution complicated by a one-way hash function and the problem by the ECDLP at the same time. Its difficulty is even harder than the ECDLP itself. Thus, the proposed scheme succeeds in withstanding the public key substitution attack.

With the warrant M_w, and proxy signer public key Q_p, the original signer Q_1 may intend to simultaneously forge his own public key Q_1 and the point R from the given proxy multi-signature public key $\overline{Q}_p$ to make the following signature verification equation certifiable:

$$\overline{Q}_p = Q_p + h(M_w, x_{Q_1}, x_{Q_p}, x_R) \times Q_1 + \cdots + h(M_w, x_{Q_n}, x_{Q_p}, x_R) \times Q_n - R \qquad (1)$$

In one case, an attacker may randomly select a point $Q_1' = (x_{Q_1'}, y_{Q_1'})$ as his public key, and then he computes the corresponding point $R' = (x_{R'}, y_{R'})$ based on the Equation (1). The difficulty is harder than that by the ECDLP. In another case, an attacker may randomly select a point $R' = (x_{R'}, y_{R'})$, and then he computes the corresponding $Q_1' = (x_{Q_1'}, y_{Q_1'})$; the difficulty is also harder than that by the ECDLP.

3.2 Performance Analyses

In order to present a contrast, the performance of the Scheme by Sun and the proposed one is formed into the following tables. Table 1 is the definitions of the given notations, and Table 2 shows the relationships of the various operations. As to the generation and verification phases, they are shown as Tables 3. Then, the required time complexities in the different phases are estimated as Tables 4, so that the efficiency in executing can be specifically analyzed.

Table 1: Definitions of Notions

Notations	**Definitions**
T_{MUL}	the time for the modular multiplication
T_{EXP}	the time for the modular exponentiation
T_{ADD}	the time for the modular addition
T_{EC_MUL}	the time for the multiplication of a number and an elliptic curve point
T_{EC_ADD}	the time for the addition of two points in an elliptic curve

Through the statements [12], the relationships of various operations can be included so as to specify the time complexity:

- g^x mod p, where p is a 1024-bit prime and x is a random 160-bit integer.
- $k \times B$ is given, where $B \in E(Z_p)$, E is an elliptic curve defined over Z_p, $p \approx 2^{160}$, and k is a random 160-bit integer.

Thus, time complexity is provided with the following relationship:

Table 2: Relationships of Various Operations

$T_{EXP} \approx 240T_{MUL}$	$T_{EC\_MUL} \approx 29T_{MUL}$	$T_{EC\_ADD} \approx 0.12T_{MUL}$	T_{ADD} is negligible

Table 3: Phases of Sun′s and Proposed Proxy Multi-Signature Schemes

Items		Scheme by Sun	Scheme by us
Key Generation	**Private Key**	s_i , s_p	d_i , d_p
	Public Key	$v_i = g^{s_i}$ (mod p), $v_p = g^{s_p}$ (mod p)	$Q_i = d_i \times B = (x_{Q_i}, y_{Q_i})$ $Q_p = d_p \times B = (x_{Q_p}, y_{Q_p})$
Sub-Delegation Parameter Generation		k_i , $K_i = g^{k_i}$ (mod p)	$R_i = k_i \times B = (x_{R_i}, y_{R_i})$ $R = \sum_{i=1}^{n} R_i = (x_R, y_R)$
		$\sigma_i = s_i \cdot v_i + k_i \cdot h(M_w, K_i)$ (mod p-1)	$s_i = d_i \cdot h(M_w, x_{Q_i}, x_{Q_p}, x_R) - k_i$ (mod t)
Sub-Delegation Parameter Verification		$g^{\sigma_i} = v_i^{v_i} \cdot K_i^{h(M_w,K_i)}$ (mod p)	$R_i^{'} = h(M_w, x_{Q_i}, x_{Q_p}, x_R) \times Q_i - s_i \times B$
Proxy Multi-Signature Secret Key Generation		$\sigma = s_p \cdot v_p + \sum_{i=1}^{n} \sigma_i$ mod (p-1)	$\overline{d}_p = d_p + \sum_{i=1}^{n} s_i$ (mod t)
Verification of the Proxy Multi-Signature		$v = v_p^{v_p} \cdot v_1^{v_1} \cdots v_n^{v_n} \cdot K_1^{h(M_w,K_1)} \cdots K_n^{h(M_w,K_n)}$ (mod p)	$\overline{Q}_p = Q_p + h(M_w, x_{Q_1}, x_{Q_p}, x_R) \times Q_1 + \cdots + h(M_w, x_{Q_n}, x_{Q_p}, x_R) \times Q_n - R$

Table 4: Time Complexity and Estimation of Proxy Multi-Signature Schemes

Items	Scheme by Sun		Scheme by us	
	Time Complexity	**Roughly Estimation**	**Time Complexity**	**Roughly Estimation**
Key Generation	(n+1)T_{EXP}	240(n+1)T_{MUL}	(n+1)T_{EC_MUL}	29(n+1)T_{MUL}
Sub-Delegation Parameter Generation	nT_{EXP}+ 2nT_{MUL}+ nT_{ADD}+ n*Hashing*	242nT_{MUL}+ n*Hashing*	nT_{EC_MUL}+ nT_{MUL}+ (n-1)T_{EC_ADD} +nT_{ADD} n*Hashing*	(30.12n+0.12)T_{MUL} + n*Hashing*
Sub-Delegation Parameter Verification	3nT_{EXP}+ nT_{MUL}+ n*Hashing*	721nT_{MUL}+ n*Hashing*	2nT_{EC_MUL}+ (2n-1)T_{EC_ADD}+ n*Hashing*	(58.24n-0.12)T_{MUL}+ n*Hashing*
Proxy Multi-Signature Secret Key Generation	1T_{MUL}+ nT_{ADD}	1T_{MUL}	nT_{ADD}	Negligible

Verification of the Proxy Multi-Signature	$(2n+1)T_{EXP}$+ $2nT_{MUL}$+ n*Hashing*	$(482n+240)T_{MUL}$+ n*Hashing*	nT_{EC_MUL} + $(n+1)T_{EC\_ADD}$ + n*Hashing*	$(29.12n+0.12)T_{MUL}$+ n*Hashing*

4 Conclusions

The research in the paper contributes a new proxy-protected proxy multi-signature scheme secure and more efficient than those by Sun. Noteworthy is that the additional demand for a secure manner in the previous related solutions, delivering the delegation parameter from the original signer to the proxy one, is simplified to be omissible in enciphering and deciphering. Especially for the proposed multi-signature scheme, it makes the computation overhead independent from the number of the original signer, so that the amount of operation for the verification can be greatly reduced. In the way, the practicability of the proxy signature techniques can be pushed ahead.

References

[1] M. Mambo, K. Usuda, and E. Okamoto, *Proxy Signatures for Delegating Signing Operation*, “Proc. 3rd ACM Conference on Computer and Communications Security,” ACM press, 1996, pp.48-57.

[2] M. Mambo, K. Usuda, and E. Okamoto, *Proxy Signatures: Delegation of the Power to Sign Messages*, “IEICE Trans. Fundamentals,” Vol. E79-A, No. 9, Sep. 1996, pp.1338-1354.

[3] L. Yi, G. Bai, and G. Xiao, *Proxy multi-signature scheme: A new type of proxy signature scheme*, “Electronics Letters,” Vol. 36, No. 6, 2000, pp.527-528.

[4] S. Kim, S. Park, and D. Won, *Proxy Signatures, Revisited*, “ICICS'97,” Lecture Notes in Computer Science 1334, Springer-Verlag, 1997, pp.223-232.

[5] H. M. Sun, *On Proxy Multi-Signature Schemes*, “Proceedings of the International

Computer Symposium," 2000, pp.65-72.

[6] H. M. Sun, *Improved Proxy Signature Schemes*, "Proceedings of the International Computer Symposium," 2000.

[7] N. Koblitz, *Elliptic Curve Cryptosystems*, "Mathematics of Computation," Vol. 48, 1987, pp.203-209.

[8] N. Koblitz, "A Course in Number Theory and Cryptography," New York, NY: Springer-Verlag, Second edition, 1994.

[9] V.S. Miller, *Uses of Elliptic Curves in Cryptography*, "Advances in Cryptology-Crypto'85, Proceedings, Lecture Notes in Compute Science, New York, NY: Springer-Verlag," No. 218, 1985, pp.417-426.

[10] S. Pohlig and M. Hellman, *An Improved Algorithm for Computing Logarithms over GF(p) and Its Cryptographic Significance*, "IEEE Transactions on Information Theory," Vol. 24, 1978, pp.106-110.

[11] *A Certicom Whitepaper, The Elliptic Curve Cryptosystem*, July 2000, http://www.certicom.com

[12] Chu-Hsing Lin and Cheng-Lung Lee, *Elliptic-Curve Undeniable Signature Schemes*, "Proceedings of the Eleventh National Conference on information Security," 2001, pp.331-338.