An Implicit ELGamal Digital Signature Scheme

Haipeng Chen
College of Computer Science and Technology, Jilin University, Changchun, China
Email: chenhp@jlu.edu.cn

Xuanjing Shen and Yingda Lv
College of Computer Science and Technology, Jilin University, Changchun, China
Email: xjshen@jlu.edu.cn; lvyingda1983@163.com

***Abstract*—This paper introduced a detail ElGamal digital signature scheme, and mainly analyzed the existing problems of the ElGamal digital signature scheme. Then improved the scheme according to the existing problems of ElGamal digital signature scheme, and proposed an implicit ElGamal type digital signature scheme with the function of message recovery. As for the problem that message recovery not being allowed by ElGamal signature scheme, this article approached a method to recover message. This method will make ElGamal signature scheme have the function of message recovery. On this basis, against that part of signature was used on most attacks for ElGamal signature scheme, a new implicit signature scheme with the function of message recovery was formed, after having tried to hid part of signature message and refining forthcoming implicit type signature scheme. The safety of the refined scheme was anlyzed, and its results indicated that the new scheme was better than the old one.**

***Index Terms*—ElGamal-type digital signature scheme; message recovery; implicit signature scheme; security analysis**

I. INTRODUCTION

ElGamal signature scheme is designed to use as a signature, and its speed of encryption and decryption is relatively slower than the symmetric algorithm, it is the common problem of all practical public key algorithms at present [1-3]. It is a non-deterministic two-key system. In terms of the same plaintext message, due to different parameters chosen randomly, it has different signatures. Most digital signature systems in the public did not have the message recovery function. Signature scheme allowing message recovery has many advantages [4-7], such as shorter signature for shorter message; meanwhile, it puts the message together with validation[4]. Nyberg and Rueppel had improved the broad-based ELGamal mode [8-10], and a series of signature schemes had been received, which could verify the signature while recovering the message.

The implicit digital signature scheme conceals part of the signature from the other data in the signature, and constitutes an implicit signature on the message by using the data of the implicit part of the signature. Signature receiver can verify the signature of the implicit signature to come to the verification of the real signature. As for the generalized ELGamal type signature scheme, Harn and Xu pointed out that there are a total of 18 security ELGamal type signature schemes [11]. Qi Ming and Xiao Guozhen promoted the Chang-Liao password authentication scheme based on generalized ELGamal type scheme in [12]. To make a signature scheme more secure, this paper puts forward an improved scheme of three implicit ElGamal type signature schemes. According to the different ways of hiding the signature, the schemes are divided into three types: Ⅰ (a with s) type, Ⅱ (b with s) type, Ⅲ (c with s) type. The feature of the improved scheme is to conceal part of the original signature s. So an attacker can not use part of signature s to substitute the usual attack and can increase the number of unknowns (key x, k, s) in original signature equation, while the forgers are not aware of these unknowns, and thus prevent the reuse of k to exposure to key x. Signature receiver can verify the signature of the implicit signature to come to the verification of the real signature. As the implicit signature scheme was improved; method of message recovery was given and it was applied to ElGamal digital signature scheme, thus an implicit digital signature scheme with the function of message recovery was formed. The analyzed results show that: the security of the improved scheme has been improved significantly.

II. ELGAMAL DIGITAL SIGNATURE SCHEME

A. General definition of digital signature scheme

Generally speaking, a digital signature [13] mainly has two algorithms. Signer can use a (secret) signing algorithm to sign a message, leading to the signature by a public verification algorithm to verify. Verification algorithm makes an answer with "real" or "false" according to whether the signature is real when given a pair of signature. A digital signature scheme can be described when meeting the following conditions (P, A, K, S, V):

(1) P is a finite set composed by all possible messages;

National Natural Science grant (No.60773098);
Scientific and Technological Development Foundation of Jilin Province(No.20080317).

doi:10.4304/jsw.6.7.1329-1336

(2) A is a finite set composed by all possible signatures ;

(3) K is a finite set composed by all possible keys, which is key space;

(4) As for each $k \in K$, there is a signature algorithm $Sig_k(\cdot) \in S$ and a corresponding verification algorithm $Ver_k(\cdot,\cdot) \in V$. Each $Sig_k(\cdot) \in S$ and $Ver_k(\cdot,\cdot) \in V : P \times A \rightarrow \{true, false\}$ is a function which satisfies the following equation: as for each message $x \in P$ and each signature $y \in A$, there is $Ver_k(x, y) = true$, If and only if $y = Sig_k(x)$, $Sig_k(\cdot)$ and $Ver_k(\cdot,\cdot)$ both are function of polynomial time. $Ver_k(\cdot,\cdot)$ is a public function, while $Sig_k(\cdot)$ is a secret function.

B. The description of the ElGamal type digital signature scheme based on discrete logarithm problem on Z_p^*

Suppose p is a intractable prime on Z_p^* the discrete logarithm problem[15-16], q is a large prime factor of p-1, or p=q, when q<p, select a element $\alpha \in Z_p^*$ of an order for the q randomly; When p=q, randomly select a element $\alpha \in Z_p^*$, or $P \in Z_p^*, A = Z_p^* \times Z_q, (q \langle p)$ or $A = Z_p^* \times Z_{p-1}(q = p)$ of an order for the p-1. Definition:

$$K = \left\{ (p, q, \alpha, a, \beta) \middle| \beta = \alpha^a (\bmod p) \right\}$$

Where p, q, α, β are public, a is private.

For $K = (p, q, \alpha, a, \beta)$ and a secret random $k \in Z_q^*(q < p)$ or $k \in Z_{p-1}^*(q = p)$, definition:

$$Sig_k(x, k) = (\gamma, \delta) \tag{1}$$

Where $\gamma = \alpha^k \bmod p (\bmod q)$.

When q<p, δ satisfies the equation:

$$k \cdot f(\gamma, x, \delta) + a \cdot g(\gamma, x, \delta) + h(\gamma, x, \delta) \equiv 0 \bmod q \tag{2}$$

When q=p, δ satisfies the equation:

$$k \cdot f(\gamma, x, \delta) + a \cdot g(\gamma, x, \delta) + h(\gamma, x, \delta) \equiv 0 (\bmod p - 1) \tag{3}$$

f, g, h is a public function, and is calculated easily from (1) and (2).

For $x \in Z_p^*, \gamma \in Z_q^*$ and $\delta \in Z_q$, definition:

$$Ver(x,\gamma,\delta) = true \Leftrightarrow \gamma^{f(\gamma,x,\delta)} \cdot \beta^{g(\gamma,x,\delta)} \cdot \alpha^{h(\gamma,x,\delta)} \equiv 1 (\bmod p)(\bmod q) \tag{4}$$

f, g, h is a public function and (x, γ, δ) is known public, so any one can verify the equation (3).

If the signature is constructed correctly, then the validation will be successful, because

$$\begin{aligned} &\gamma^{f(\gamma,x,\delta)} \cdot \beta^{g(\gamma,x,\delta)} \cdot \alpha^{h(\gamma,x,\delta)} (\bmod p)(\bmod q) \\ &= \alpha^{kf(\gamma,x,\delta)} \alpha^{ag(\gamma,x,\delta)} \alpha^{h(\gamma,x,\delta)} (\bmod p)(\bmod q) = 1 \end{aligned} \tag{5}$$

When f, g, h take a different function, different digital signature scheme will be gotten, and we referred to this type of scheme as the ElGamal digital signature scheme.

In the above scheme,

$$q = p,\ f(\gamma, x, \delta) = \delta, g(\gamma, x, \delta) = \gamma, h(\gamma, x, \delta) = -m \tag{6}$$

When taking the above-type, the scheme is the ElGamal digital signature scheme. At this point, the signature algorithm and verifying algorithms of the scheme change correspondingly:

$$Sig_k(x, k) = (\gamma, \delta), \gamma = \alpha^k \bmod p, \delta = (x - a\gamma)k^{-1} \bmod p - 1 \tag{7}$$

And

$$Ver(x, \gamma, \delta) = true \Leftrightarrow \beta^{\gamma} \gamma\delta \equiv \alpha^x (\bmod p) x, \gamma \in Z_p^*, \delta \in Z_{p-1} \tag{8}$$

C．Analysis of ElGamal Digital Signature Algorithm Security

In general, the following are the main ways of attack: a direct hack on the private key[14-16].

1) Following the launch of RSA in 1978, has spent much effort to find the defects that can be deciphered. Although it is used within a certain scope of the agreement is not without risk, the algorithm's basic security is guaranteed. But ELGamal algorithm has not a decoding test of detailed cryptanalysis , there are serious technical defects.

2) For activities attacks and counterfeit attacks is fragile. If an attacker successfully replaces a legitimate user's public key by using private key that is randomly selected corresponding to public key, then user will be able to forge signatures.

3) The substitution attack. These attacks include the use of some of the signatures s and only use the public key Y. The substitution attack carried out by using some of the signature s is major attack that the signature program ELGamal face to.

4) The forgery attack. Starting from the signature,forgers make any changes to form the signature of another message m, which is possible to meet the same verification equation.

5) Random prive key k can not be repeatedly used to sign different messages. Otherwise, an attacker can easily obtain the signer private key x.

6) He and Keisler point out that can forge signer sign any message. If three random key k, (i = 1,2,3) satisfy the k3 = k1 + k2, then r, (i = 1,2,3) to meet the r3 = r1r2. Then the attacker can obtain the key x. This is similar to the homomorphism attack that is faced by RSA , the difference is, the homomorphism attack of RSA signature is only used to forge signature , and can be overcome by using hash function .while the homomorphism attack of the ELGamal digital signature scheme, has yet to find effective solutions by now.

7) Subliminal channel problem. Closed threshold on the lower channel of ELGamal digital signature scheme ,has not seen any study findings so far.

According to the above analysis on ElGamal Digital Signature Algorithm, due to the discrete logarithm problem has yet no possible solution has been worked out yet, so the ElGamal type digital signature algorithm based on the question has high security keys. Before the discrete logarithm problem is effectively resolved, any direct attack on the keys, the computational needs are staggering[17-18].

According to the signature and the verify equation, the signer needs to complete a signature through a power operation and an inverse operation, and the verifier needs three power operations. Given that sign documents or information for the sign may be more, but the verifier is the different user; this conditions which the computational complexity of verifier is higher than the computational complexity of signers.

III. The Implicit Signature Scheme with Message Recovery

Setting p as a large prime number, which makes calculating the discrete logarithm of p a hard problem. Moreover, p-1 contains large prime. Then select randomly a element $\alpha \in Z_p^*$ with an order for p-1. $P = Z_p^*$ is a set of all possible messages. While $A = Z_p^* \times Z_p^* \times Z_{p-1}$ is a set [21] of all possible signatures, definition:

$$K = \left\{(p,\alpha,x,y) \middle| y = \alpha^x \bmod p\right\}$$

Where parameters p , y and α are known to every user. Signer's private key x is element belonging to set Z_{p-1}^*, the corresponding public key generated by x is:

$$y = \alpha^x \bmod p \tag{9}$$

Sign a message m, which is an element on Z_p^*.For

$$K = \{(p,\alpha,x,y) | y = \alpha^x \bmod p\}$$

and a secret random number $k,t \in Z_{p-1}$, define:

$$Sig_k(m,k) = (r,i,v)$$

Where r is calculated by

$$r = m\alpha^{-k} \bmod p \tag{10}$$

and assume

$$\gcd(ry^{'}, p-1) = 1 \tag{11}$$

Where “gcd” is greatest common factor. If the above equation does not hold, we can fixe the k and choose another value of t to make it hold.

Then calculation of s is defined by the following equation:

$$s = (1+kr)x^{-1} \bmod (p-1) \tag{12}$$

Among them

$$sx - kr = 1 \bmod (p-1) \tag{13}$$

is signature equation. Then calculate:

$$u = y^t (\bmod p) \tag{14}$$

And

$$v = t + su \bmod p-1 \tag{15}$$

The message signature is (r,u,v), for

$$m \in Z_p^*, r \in Z_p^*, u \in Z_p^* \ and \ \ v \in Z_{p-1}$$

Define:

$$Ver(r,s) = TRUE \Leftrightarrow y^v (r^r \alpha)^{-u} = m^{-ur} u \bmod p$$

When validating and recovering messages, firstly, receiver calculates:

$$c = y^v (r^r \alpha)^{-u} (\bmod p) \tag{16}$$

and then determine whether c satisfies the condition (17) to verify signature.

$$c = m^{ur} u \tag{17}$$

Theorem 3.1: In the above signature scheme, we assume $\gcd(ry^t, p-1) = 1$ is reasonable; namely, the t which can satisfy the equation exists, and the choice of t be completed in limited steps.

Prove: When t takes over each element of Z_{p-1}, y' takes over each element of Z_p^*. Since there are $\phi(p-1)$ elements relatively prime with p-1 in Z_p^*, then there are also $\phi(p-1)$ elements prime with p-1 in $\left\{ry^t \middle| t \in Z_{p-1}\right\}$, that is, t which satisfies $\gcd(ry^t, p-1)=1$ exists, and the number is $\phi(p-1)$.

As t is a random selection in Z_{p-1} to satisty the $\gcd(ry^t, p-1)=1$, then we can know the expected setps in selecting such t, the expectation is $p-1/\ \phi(p-1)$.

The proof has been completed.

Theorem 3.2: when signing message *m*, if its signer complied with the steps in the above mentioned scheme, it can be verified that signature (r,u,v) is true signature of the message.

Prove: From (16), we can get:

$$\begin{aligned} c &= y^v (r^r \alpha)^{-u} \bmod p \\ &= y^{t+su} (m^r \alpha^{kr} \cdot \alpha)^{-u} \bmod p \\ &= u\alpha^{xsu} m^{-ur} \alpha^{-kru} \alpha^{-u} \bmod p \\ &= u\alpha^{(xs-kr-1)^u} m^{-ur} \bmod p \end{aligned}$$

Because $xs-kr=1 \bmod p-1$, then $c = u\cdot m^{-ur} \bmod p$, that is to say signature (r,u,v) is true signature of the message.

The proof has been completed.

IV. Recovery of Implicit Signature Message

In (17) the value of message *m* is unknown, so, when verifying signatures we have to recover message *m*. Euclidean algorithm[22] for polynomial[7] is used to recover message *m*.

First, define polynomial

$$P(X) = X^r - c\alpha^{-1} \bmod p \qquad (18)$$

Where, message *m* is one root of polynomial $P(X)$. Because *r* and *p*-1 are co-prime, there is only one root for polynomial $X^r - c\alpha^{-1} \bmod p$ on limited domain Z_p, and the root is *m*.

Theorem 4.1: Recover message from signature (r,u,v), and message m is given though following equation:

$$\begin{aligned} m &= (c^{-1}u)^t \\ &= \left[y^{-vr^{-1}} r(\alpha\cdot u)^{r^{-1}} \right]^{u^{-1}} \bmod p \end{aligned} \qquad (19)$$

Prove: Noted m satisfies the equation

$$m^{ur} = c^{-1}u \bmod p$$

Since ru and p-1 are co-prime, then choose t and make it like this:

$$rut = 1(\bmod p-1) \qquad (20)$$

Then:

$$\begin{aligned} (c^{-1}u)^t &= \bmod p \\ &= m^{urt(\bmod p-1)} \bmod p \\ &= m \end{aligned}$$

So:

$$\begin{aligned} m &= (c^{-1}u)^t \\ &= \left[y^{-vr^{-1}} r(\alpha\cdot u)^{r^{-1}} \right]^{u^{-1}} \bmod p \end{aligned}$$

Thus, we recover the message m.

The proof has been completed.

Theorem 4.2: Message *m* is the number on the contrary of the constant term of the greatest common factor of polynomial $P(X) = X^r - c\alpha^{-1} \bmod p$ and polynomial $X^p - X$, that is:

$$X - m = \gcd(X^r - c\alpha^{-1}, X^p - X) \qquad (21)$$

where "gcd" is greatest common factor.

Prove: there are three steps for the proof.

1) Message *m* is one root of polynomial:

$$P(X) = X^r - c\alpha^{-1} \bmod p.$$

Because

$$c = \alpha\cdot m^r (\bmod p)$$

So

$$m^r = c\alpha^{-1} (\bmod p)$$

that is

$$m^r - c\alpha^{-1} = 0 (\bmod p) \qquad (22)$$

so message *m* is one root of polynomial $P(X)$.

2) There is only one root for polynomial $P(X) = X^r - c\alpha^{-1} \bmod p$ on Z_p.

It can be divided into the following two cases for discussion:

a) When $c\alpha^{-1} = 1$, that is $X^r - 1 = 0$. Equation $X^{(r,p-1)} - 1 = 0$ can be derived from $X^{p-1} - 1 = 0$. Because $(r, p-1) = 1$ and $x-1=0 \Rightarrow x=1$, there is $x \equiv 1$, that is to say there is only one root for $X^r - 1 = 0$ on Z_p, and the root is 1.

b) When $c\alpha^{-1} \neq 1$, establish the mapping $f: x \rightarrow x^r$, which is from Z_p to Z_p^r. For any two elements $x_1, x_2 \in Z_p$, if there is $x_1^r = x_2^r$, $\left(\frac{x_1}{x_2}\right)^r - 1 = 0(\bmod p)$ can be gained. And from a), we can find out $x_1 = x_2$, so the mapping $f : x \to x^r$ is a bisection, that is to say there is only one root for $X^r - c\alpha^{-1} = 1(\bmod p)$ on Z_p.

3) Prove the establishment of equation $X - m = \gcd(X^r - c\alpha^{-1}, X^p - X)$•

On the limited domain, there is an equation:

$$X^p - X = \prod_{i=0}^{p-1}(X - x_i),\ \ x_i \in Z_p \tag{23}$$

There is no repeated factorization in $X^p - X$, but from a) it can be known that $X^r - c\alpha^{-1}$ is the r-th power repeated factorization of $X - m$, so:

$$\partial^\circ \gcd(X^r - c\alpha^{-1}, X^p - X) \le 1. \tag{24}$$

Because $m \in Z_p^*$, there is i making $x_i = m$, where $0 \le i \le p$, that is

$$X - m = \gcd(X^r - c\alpha^{-1}, X^p - X) \tag{25}$$

In summary, the theorem has been proved.

In terms of Euclidean algorithm on polynomials, we can get the following conclusions [8]: Assumed polynomial applies to polynomials f and g, and $\deg f \ge \deg g = d$ then there are:

1) The most steps of decomposition are d t, clearly every step in the process of price reduction may reduce the order for1 at least.

2) Steps required by decomposition are just d, when f and g are polynomial of recursive sequence of polynomials, defined as follows:

$$\{F_0 = 1,\ F_1 = X,\ F_{n+1} = XF_n + F_{n-1}, n \ge 1 \tag{26}$$

3) If the g is uniformly distributed in polynomial and make $f \bmod g$ be uniform distribution as well, then the number of steps of decomposition is $(1-1/p)d$.

For general polynomials, this algorithm needs large amount of computing. However, applying Euclidean algorithm for polynomial to $(X^p - X, X^r - c\cdot\alpha^{-1})$, can reduce the amount of computing largely. Where there is no repeated factorization in polynomial $X^p - X$, its roots traversal every element in Z_p . Especially $X^r - c\cdot\alpha^{-1}$ is the r-th power repeated factorization, having only one factor. This makes the amount of computing in the process of algorithm decomposition be reduced largely, and algorithm speed be faster.

According to the difference of the hidden part of the signature in the scheme, we can get three kinds of implicitly corresponding signature scheme with the function of message recovery. Signature of each scheme is (r, u, v). The lists are as follows:

Tabel 1 a inclued s concealed type signature scheme

Signature equation	Signature vefication
$r = m\alpha^k \bmod p$ $sx = rk + 1 \bmod p-1$ $u = y^t \bmod p$ $v = t + su \bmod p-1$	$c = y^v(r^r\alpha)^{-u} \bmod p$ If $c = m^{-ur}u$,then accept signature

Tabel 2 b inclued s concealed type signature scheme

Signature equation	Signature vefication
$r = m\alpha^k \bmod p$ $rx = sk + 1 \bmod p-1$ $u = r^t \bmod p$ $v = t + su \bmod p-1$	$c = r^v(\alpha^{-1}y^r)^{-u} \bmod p$ If $c = m^{-ur}u$,then accept signature

Tabel 3 c inclued s concealed type signature scheme

Signature equation	Signature vefication
$r = m\alpha^k \bmod p$ $x = rk + s \bmod p-1$ $u = \alpha^t \bmod p$ $v = t + su \bmod p-1$	$c = \alpha^v(r^{-r}y^1)^{-u} \bmod p$ If $c = m^{-ur}u$,then accept signature

V. Security Analysis

The security of the above scheme mentioned is based on the discrete logarithm problem. Assuming that the attacker does not know key *x*, digital signature of a given message *m* is going to be forged now. If the attacker chooses a value *s* first, and then finds a corresponding *r* strongly, the transcendental equation $r\log\frac{m}{r} = \log y^s\alpha^{-1}$ must be solved to find out *r*, and it is associated with the discrete logarithm. If the attacker chooses a value *r* first, and then finds a corresponding *s* strongly, the transcendental equation $s = \log\left[\alpha(\frac{m}{r})^r\right]/\log y$ must be solved. The implicit signature scheme hides part of signature s among other data, and constitutes an implicit signature on the message using the data of the hidden part of the signature; finally verifies the real signature through the implicit signature verification. Increase the number of the original signature equation unknown (key x, k, s to the forger not know), preventing key x from espousing for the reuse of k, and security of other aspects has been strengthened.

In this paper, we make the security analysis to the implicit scheme with the function message recovery by only taking I (a with s)-type scheme as an example,

1) (r,u,v) is an implicit signature of message, denoted $csig(m)$. In practice, the time can be introduced to the signature, the signature equation:

$$v = t + su \bmod p - 1$$

Modified it to:

$$v = t + sf(u,T) \bmod p - 1 \qquad (27)$$

In equation, One-way function f is introduced to prevent the attacker changing the time, this time $csig(m) = (r,u,v,T)$. This is more realistic.

In some special cases, the signature debit not only to verify the correctness of the signature, but also to verify the signature is valid. In general, there are signatures associated with the signature time, signature and its corresponding time are equally important and need to be verified. Implicit scheme has the function of verifying the signature and signature time at the same time.

2) For the given message $\{m_i, i = 1,2,\cdots,n\}$ and its correspondingly implicit signature $\{(r_i,u_i,v_i).i = 1,2,\cdots,n\}$. To obtain the signer's key x, attacker can start from the signature equation as likely treate the ELGamal type signature scheme.

Attackers try to solve the key x from equations:

$$\begin{cases} s_1x = r_1k_1 + 1 \bmod p - 1 \\ s_2x = r_2k_2 + 1 \bmod p - 1 \\ \vdots \\ s_nx = r_nk_n + 1 \bmod p - 1 \end{cases} \qquad (28)$$

and

$$\begin{cases} v_1 = t_{11} + s_1u_1 1 \bmod p - 1 \\ v_2 = t_2 + s_2u_2 \bmod p - 1 \\ \vdots \\ v_n = t_n + s_nu_n \bmod p - 1 \end{cases} \qquad (29)$$

We discuss the points from the following four cases:

a) Random key k_i and t_i were not being reused: at this time, the number of unknowns are more than the number of equations in the two equations, and the key x has no unique solution.

b) k_i is only reused :

At this time, $k_1 = k_2 = \cdots = k_n = k$.

From $r_i = m_i\alpha^{k_i} \bmod p$, we can introduce:

$$\frac{r_1}{m_1} = \frac{r_2}{m_2} = \cdots \frac{r_n}{m_n} \qquad (30)$$

As it is hard to solve $s_i(i = 1,2,\cdots,n)$ from the second equations, therefore, even if the attacker knows k_i is used repeatedly from (30), that are , it will still not solve the key x due to the number of unknown in the first equations (31) bigger than the number of equation.

$$\begin{cases} s_1x = r_1k + 1 \bmod p - 1 \\ s_2x = r_2k + 1 \bmod p - 1 \\ \vdots \\ s_nx = r_nk + 1 \bmod p - 1 \end{cases} \qquad (31)$$

c) t_i is only reused :

At this time, $t_1 = t_2 = \cdots = t_n = t$, which lead $u_1 = u_2 = \cdots = u_n = u$. If $s_i(i = 1,2,\cdots,n)$ is different form each other which is produced by the first equations, then due to the above reason, the attack is hard to solve $s_i(i = 1,2,\cdots,n)$ from the second equations, therefore ,the key x is not obtain.

$$\begin{cases} v_1 = t + s_1u \bmod p - 1 \\ v_2 = t + s_2u \bmod p - 1 \\ \vdots \\ v_n = t + s_nu \bmod p - 1 \end{cases} \qquad (32)$$

Even if there are several same $s_i(i = 1,2,\cdots,n)$, if time is introduced to signature like equation (32), the signature changes to $v = t + sf(u,T) \bmod p - 1$, then v_i are different from each other due to the different t_i . So, attackers can not judge whether s_i is same from the signature $csig(m_i)(i = 1,2,\cdots,n)$.

d) both k_i *and* t_i are reused :

From the above analysis we can know both k_i and t_i are reused, which won't expose the key x under the condition that $s_i(i = 1,2,\cdots,n)$ is different from each other.

In a word, using the ElGamal signature scheme is not same to using the implicit signature scheme which can not be strict greatly in the selection of random key, but relatively flexible.

3) From the above security analysis, we can see that there are also with the following state attacks for the ELGamal signature scheme with the function of message recovery:

If the three random keys $k_i, i = 1,2,3$ is calculated $r_i(i = 1,2,3)$, which satisfies $k_3 = k_1 + k_2$, then that is $\frac{r_1r_2}{m_1m_2} = \frac{r_3}{m_3}$.

So an attacker can solve key x using these correlations solutions. The solving process can be obtained through the signature equation

$$xs_i + k_i r_i = 1 \bmod p-1, i = 1,2,3$$

and relationship equation $k_3 = k_1 + k_2$ to obtain the following equation:

$$(1-xs_1)r_1^{-1} + (1-xs_2)r_2^{-1} = (1-xs_3)r_3^{-1} \Rightarrow x =$$
$$(r_1^{-1} + r_2^{-1} - r_3^{-1})(s_1 r_1^{-1} + s_2 r_1^{-1} + s_2 r_1^{-1} - s_3 r_3^{-1})^{-1}$$

Thus, key x has been solved, and the system has been deciphered.

As for the implicit signature scheme, since $s_i (i = 1,2,3)$ is not public, no one can obtain the key x using the above method.

In addition, when $t_i (i = 1,2,3)$ satisfy $t_3 = t_1 + t_2$, attacker can not obtain anything from the equations:

$$\begin{cases} v_1 = t_1 + s_1 u_1 \\ v_2 = t_2 + s_2 u_2 \\ v_3 = t_3 + s_3 u_3 \end{cases} \qquad (33)$$

Therefore, attack with the state is hold only with ELGamal type signature scheme, but dose not hold to the implicit ELGamal type signature. In this way, we have solved the question by He and Keisler but not able to resolve [23-24].

As a deformation of ELGamal type signature scheme, the security of implicit signature is also based on the difficulty of discrete logarithm. Literature [25] showed that if b is set to the bit of the prime modulus p, then the computational complexity of requesting x from y, requesting k from r or requesting t from u is $O(\exp\sqrt{cb1nb}, \; where \; c \in (0,1)$. It is generally believed that the computational complexity of the algorithm is $O(\exp\sqrt{cb1nb})$ which is called a sub-exponential time algorithm. Therefore, as long as the scale of b (eg b come to 1024-bit) is expanded appropriately, it will be very difficult for forgers who want to solve discrete logarithm to decipher implicit signature schemes or ELGamal type signature scheme. As the complexity of the mode index problem is O (n), so the signer can easily calculate the value of y, u and r[26].

Due to structural features of implicit signature equation, the signature speed is slower one time than ELGamal signature scheme, which is the major weakness of implicit scheme. However, the calculation of u has nothing to do with the message m, it can be carried out and calculation to make up the weakness of the speed. Enhanced ELGamal type signature's speed and verification speed are relatively slower than the original ELGamal signature scheme. With the increased security, signature speed and verification speed will be subject to different influence.

VI. CONCLUSION

Signature scheme allowing message recovery has many obvious advantages such as shorter signature for short message, lower computation work for the combination of message and its signature to be sent, and so on. Most signature schemes including ELGamal signature mode do not allow message recovery. Based on ELGamal signature mode, an improved scheme was proposed, which allows message recovery and has better security than the original one. In particular, it can resist homomorphism attack and substitution attack using partial signature. This function is not available in the other ELGamal improved schemes.

Due to some features of attacks suffered by ELGamal signature, hash function must be used to ensure its security. As for ELGamal type signature scheme not using hash functions, Yen and Laib actually made a lot of efforts, but were unable to find [27]. These schemes are attacked by using some of the signatures s offensive and just carried out a public key y. Although the implicit signature program is also not completely free to use a hash function, one case of a major offense can be prevented at least without the use of hash functions . Harn and Xu pointed out that there are 18 the secure ELGamal type signature schemes, fowling attack methods of modeled Nyberg and Rueppel, you can use part of signature to substitution attack on almost 18 different schemes. If the pure public-key attach was not considered, such scheme can still maintain its security without using hash function.

REFERENCES

[1] CHEN Zhi-ming. An inproved encryption algorithm on ELGamal algorithm[J]. Computer Applications and Sostware, 2005, 22 (2): 82-85.

[2] Wang Li, Xing Wei, Xu Guang-zhong. ElGamal public-key cryptosystem based on integral quaternions[J]. Computer Applications, 2008,28(5):1156-1157.

[3] Lu Hong-wen, Sun Yu-hua. A Public-key Cryptography Using Integral Quaternions[J]. Journal of Tong Ji University, 2003, 31(12).

[4] HUANG Zhen-jie, WANG Yu-min, CHEN Ke-fei. Generalization and improvement of Nyberg-Rueppel message recovery blind signatures[J]. Journal on Communications, 2005 , 26(12): 131-135.

[5] CHEN Hui-yan, LB Shu-wang, LIU Zhen-hua . Identity Based Signature Scheme with Partial Message Recovery [J]. ChineseJournal of Computers, 2006, 29 (9) : 1622-1627 .

[6] Cao Tian-jie, Lin Dong-dai. Security analysis of a signature scheme with message recovery[J]. Journal of Zhejiang University(Science Edition) , 2006,33 (4) :396～397

[7] Kan Yuan-ping. A Signat ure Scheme wit h Message Recovery Based on Elliptic Curves[J]. Computer engineering and science, 2010, 32(2):58-59.

[8] Yberg,K.and Rueppel,R.A. ”message recovery for signature schemes based on the discrete logarithm problem,” in EUROCRYPT,1994, 182~193.

[9] Wang Qing-ju, Kang Bao-yuan, Han Jin-guang.. Several New ELGamal Type Digital Signature Schemes and Their Enhanced Schemes[J]. Journal of East China Jiaotong University, 2005, 22(5): 127-138

[10] Zhang Hui-ying, Zhang Jun. Research and Design of an Improved ELGamal Digital Signature Scheme[J]. Computer Engineering and Scinece, 2009, 31(12): 35-38.

[11] Ham, L. and Xu,Y. Design of generalized ElGamal type digital signature scheme based on the discrete logarithm.Electronic Letters, 1994.31(24).

[12] QI Ming, Xiao Guo-zhen. Security and performance analysis of two kinds of ELGamal type digital signature algorithm[J]. Journal of Electronica Science, 1997, 19(3):346~349.

[13] ELGAMAL T. A public key cryptosystem and a signature scheme based on discrete logarithms[J]. IEEE Trans Inform Theory.1985,31(4): 469-472.

[14] WANG Li, XING Wei, XU Guang-zhong. ElGamal public-key cryptosystem based on integral quaternions[J]. Journal of Computer Applications, 2008, 28(5):1156-1157.

[15] D. Chaum, C. Cr´ epeau, and I. Damg˚ ard, “Multiparty unconditionally secure protocols,” STOC ’88.

[16] Yiannis Tsiounis, Moti Yung. On the Security of ELGamal Based Encryption[J]. Computer Science, 1998. Vol.1431: 117-134

[17] C.P. Schnorr and M. Jakobsson : Security of Discrete Log Cryptosystems in the Random Oracle and Generic Model. TR report University Frankfurt and Bell Laboratories 1999.

[18] C.P. Schnorr and M. Jakobsson : Security of Discrete Log Cryptosystems in the Random Oracle and Generic Model. TR report University Frankfurt and Bell Laboratories 1999.

[19] Kaisa Nyberg and Rainer A.Rueppel. Message Recovery for Signature Schemes Based on the Discrete Logarithm Problem[J]. Designs,Codes and Cryptography, 1996, 7:61-81.

[20] Changshe Ma, Jian Weng, Yingjin Li, and Robert Deng. Efficient discrete logarithm based multi-signature scheme in the plain public key model[J]. Designs,Codes and Cryptography, 2010,54(2): 122-133.

[21] WANG Shu-hong, WANG Gui-lin, BAO Feng, et al. Cryptananlysis of a proxy blind signature scheme based on DLP[J]. Journal of Software, 2005, 16(5): 911-915.

[22] YUAN Dan-shou, RONG Meng-tian. A scalable architecture for inversion based on modified Euclid's algorithms[J]. Journal of Shanghai Jiaotong University, 2005, 40(1): 36-40.

[23] Knuth, D.E, The art of computer programming. Vol 2: SeminumericalAlgorithm(AddisonWesley,Reading,Mass,1981) , 2nd Edn.

[24] S. Pohlig and M. Hellman. An improved algorithm for computing logarithms over GF(p) and its cryptographic significance. IEEE Transaction on Information Theory IT-24(1988).

[25] Yen, S. M. and Laih, C. S. New digital signature scheme based on discrete logarithm. Electronic Letters, 1999

[26] ZHANG Qing-po, CHEN Cai-yun, CHEN Lu-sheng, et al. EIGamal cryptosystem and digital signature scheme based on polynomials over finite fields. JOURNAL ON COMMUNICATIONS, 2005, 26(5): 69-72.

[27] He, J. and Keisler, T. Enhancing the security of ElGamal’s signature scheme. IEEproc. Comput.Digit. Tech, 1994,141(4);249~252.

Haipeng C. Chen, male, was born in Cao County, Shandong, June, 1978. He received bachelor degree in 2003 and master degree in 2006 both from Jilin University.

Now he is a lecturer and a Ph.D candidate in the college of computer science and technology, Jilin University. His research interests are computer network security, digital image processing and pattern recognition.

Dr. Chen is membership of China Computer Federation (E20-00 15167M).

Xuanjing B. **Shen**, male, was born in Helong County, Jilin Province, December, 1958. He received bachelor degree in 1982, master degree in 1984, and PhD degree in 1990 all from Harbin Institute of Technology respectively.

He is a professor and PhD supervisor currently in the college of computer science and technology, Jilin University. His research interests are multimedia technology, computer image processing, intelligent measurement system, optical-electronic hybrid system, and etc.

Yingda A. **Lv**, female, was born in Wenan County, Hebei Province, January, 1983. She received bachelor degree in 2007 and master degree in 2010 from Jilin University.

Now she is a Ph.D candidate in the college of computer science and technology, Jilin University. Her research interests are digital image processing and pattern recognition.

Dr. Lv is student membership of China Computer Federation (E20-00 15161G).