A Distributed E-Business System Based on Conic Curve

Xinxia Song, Zhigang Chen
Zhejiang Wanli University
No.8 South Qian Hu Road Ningbo
CHINA
http://www.zwu.edu.cn

Abstract: - A distributed E-Business System based on conic curve is proposed. This scheme is composed of two parts, constructing license and validating license. Because the security of license is determined by private key, not the arithmetic itself, user can not construct new license by given license and the public key as long as the private key is not leaked. Since encoding and decoding over conic are easily implement on conic curves, it has enabled our scheme to greatly enhance efficiency. We also analysis its security. The entire process guarantees the security and reliability.

Key-Words: - conic curve, E-Business system, public-key cryptosystem, digital signature, proxy signature

1 Introduction

The rapid development of Internet is pushing the e-Business to go to the front stage. As a consequence, the electronic registration on Internet also becomes more diffused, and it can apply to many aspects, for example, the registration of online banking, software, etc. The electronic registration on Internet can reduce cost, and also the license is very easy to transmit via Internet and to verify (compare with disk and disk fingerprint). Therefore, it is more and more popular for people to use Internet electronic registration. However, the most important issue of Internet electronic registration is security problem .

The notion of a threshold signature scheme has been extensively studied. A k out of l threshold signature scheme is a protocol that allows any subset of k players out of l to generate a signature, but that disallows the creation of a valid signature if fewer than k players participate in the protocol. This nonforgeability property should hold even if some subset of less than k players are corrupted and work together. For a threshold scheme to be useful when some players are corrupted, it should also be robust, meaning that corrupted players should not be able to prevent uncorrupted players from generating signatures.

Nowadays, two main hard arithmetic problems are used in public key encryption, namely, the integer factorization problem and the discret logarithm problem. Among all public key schemes, the only ones with existing commercial realizations are RSA or Rabin-Williams schemes, related to the factoring problem, and ElGamal scheme, related to the discret logarithm problem.

In this paper, we propose a distributed electronic authentication scheme based on conic curves. Our scheme based on conic curves. In 1998, after Zhang designed a conic group in literature [1], Cao creatively presented the concept of conic curve cryptography in [2]. Later Cao proposed a conic analog of RSA cryptosystem and some improved RSA cryptosystems in [3]. A important conclusion about cryptosystem based on conic curves in [4] is that the efficiency and the security of the public key cryptosystem based on the DLP in conic curve groups are not stronger than those based on the DLP in finite fields. But an exciting characteristic of conic is both encoding and decoding over conic are easily implemented. As an alternative algebra curve technology, we believe conic deserves the further study in cryptography.

An efficient (k, n) threshold ElGamal type public key cryptosystem was shown by Desmedt and Frankel [5] such that

(1) (k, n) members must cooperate to decrypt a ciphertext.

(2) Any $k-1$ dishonest members cannot decrypt any ciphertext.

This system requires a trusted center. Hwang [6] and then Pedersen [7] showed that the trusted center can be eliminated. In the system of Hwang [6], however, the size of the group public key is much larger than that of Desmedt and Frankel [5] because each member publicizes his own public key. In the system of Pedersen [6], the public key is as small as that of Desmedt and Frankel [5]. Pedersen's system makes use of a noninteractive verifiable secret sharing scheme [8]. Desmet and Frankel [9] showed

a (*k, n*) threshold RSA type digital signature scheme such as

(1) (*k, n*) members must cooperate to issue a signature.

(2) Any *k-1* dishonest members cannot forge a signature.

This scheme required a trusted center. Park and Kurosawa [10] showed a (*k, n*) threshold ElGamal type digital signature scheme which requires no trusted center. The ElGamal type digital signature which is applicable to this scheme is composed of only a linear combination of shared secrets. This scheme, however, requires a enciphered communication between signers when they communicate across a network.

In this paper, we propose a public key cryptosystem scheme on conic curves over the ring Zn. Our scheme is motivated by KMOV scheme, but it remove some restrictive condition from KMOV scheme and constructed on conic curves. Its security bases on the difficulty of factoring a composite number n, just like RSA. It can resist some of the known attacks on RSA. We constructed digital signature and a proxy signature on our scheme.

The remainder of the paper is organized as follows. Section 2 gives a short introduction to conic curves over a finite field. In section 3, we show some properties of conic curves over a ring, which are used in the succeeding sections. Section 4 gives a introduction to conic curve digital signature algorithm. Section5 propose a threshold signature scheme based on conic curve. Section 6 propose a Proxy Signature Scheme based on conic curve. Section 7 proposes a public ke y cryptosystem scheme on conic curves over the ring Zn and describes the signature scheme. Section 8 present a recognizable distribution scheme based on conic curve. Section 9 propose a distributed E-Business System and discusses the security of the proposed scheme.

2 Conic Curves over a Finite Field

Let p be an odd prime and F_p be a finite field of p elements. Let F_p^* be tile multiplication group of F_p. Then, without loss of generality, we can assume

$$F_p = \{0, 1, \cdots, p-1\},$$
$$F_p^* = F_p \setminus \{0\}$$

Let us further consider the conic over an affine plane $A^2(F_p)$,

$$C(F_p): \; y2 = ax^2 - bx,\ a, b \in F_p^* \quad (1)$$

Obviously, when $x=0$, we have the origin $O(0,0)$. If $x \neq 0$, let $t = yx^{-1}$ and fill $y=xt$ in the equation (1). Then, we get

$$x(a - t^2) = b,\ a, b \in F_p^* \quad (2)$$

If $a = t^2$, the equation (2) doesn't hold; If $a \neq t^2$, from the equation (2), we will have

$$x = b(a - t^2)^{-1}, \quad y = bt(a - t^2)^{-1} \quad (3)$$

where $a, b \in F_p^*$ and $(\,)^{-1}$ denotes the multiplication inverse in F_p^*. For any $t \in F_p$ and $t^2 \neq a$, let $p(t)$ be the point (x, y) over $C(F_p)$ established by the equation (3). Moreover, an ideally defined point O, namely the point at infinity $P(\infty)$, is also recognized as a point over $C(F_p)$. Let

$$H = \{ t \in F_p ; t^2 \neq a \} \cup \{\infty\}$$

then, P: $H \rightarrow C(F_p)$ is a one-to-one map. According to [4], let us define the addition $\oplus$ of elements in $C(F_p)$. $\forall P(t) \in C(F_p)$ and $t \in H$, such that

$$P(t) \oplus P(\infty) = P(\infty) \oplus P(t) \quad (4).$$

Assume $P(t_1), P(t_2) \in C(F_p)$, where $t_1, t_2 \in H$ and $t_1, t_2 \neq \infty$, such that

$$P(t_1) \oplus P(t_2) = P(t_3) \quad (5)$$

Where

$$t_3 = \begin{cases} (t_1 t_2 + a)(t_1 + t_2)^{-1}, t_1 + t_2 \neq 0, \\ \infty, t_1 + t_2 = 0. \end{cases}$$

Obviously, $t_3 \in H$ and operation $\oplus$ is commutative. Any $P(t) \in C(F_p)$, negative element

$$-P(\infty) = P(\infty), \quad -P(t) = P(-t) \quad (6)$$

And then, from (4) ~ (6), we can easily prove $\forall P(t_1), P(t_2), P(t_3) \in C(F_p)$,

$$(P(t_1) \oplus P(t_2)) \oplus P(t_3) = P(t_1) \oplus (P(t_2) \oplus P(t_3)) \quad (7)$$

Therefore,$(C(F_p), \oplus, P(\infty))$ is a finite abelian group. And $|C(F_p)|$ can be defined as,

$$\left|C(\mathbb{F}_p)\right| = \begin{cases} p-1, (\frac{a}{p}) = 1, \\ p+1, (\frac{a}{p}) = -1. \end{cases}$$

where $(\frac{a}{p})$ is Legendre Symbol.

An exciting characteristic of conic is both encoding and decoding over conic are easily implemented. Denote $H \setminus \{\infty\}$ as H^*, and assume a message $m \in H^*$, let's demonstrate how to code it.

Encoding:

$$P(m) = (X_m, Y_m),$$
$$\begin{cases} X_m = b(a - m^2)^{-1} (\mathrm{mod}\, n) \\ Y_m = bm(a - m^2)^{-1} (\mathrm{mod}\, n) \end{cases}$$

Decoding:

$$m = Y_m \cdot X_m^{-1} \ (\mathrm{mod}\ p)$$

3 Conic Curves over the ring Z_n

We now consider conic curves over the ring Z_n, where n is an odd composite squarefree integer.

Similar to the definition of $C_p(a,b)$, an conic curve $C_n(a,b)$ can be defined as the set of pairs $(x,y) \in Z_n^2$ satisfying $y2 \equiv ax^2 - bx \pmod n$. Obviously, $O(0,0) \in C_n(a,b)$. Accord to [10], all the points of $C_n(a,b)$ can be obtained by $C_p(a,b) \times C_q(a,b)$, hence the order of $C_n(a,b)$ can be obtained by the use of $C_p(a,b)$ and $C_q(a,b)$. We have:

Proposition 1: If (a/p)= -1, $|C_n(a,b)|=(p+1)(q+1)$.

The proof we refer to [10].

By the map ϕ in [10], the add operation is defined by using of the add operation of conic curve on finite field F_p , i.e. $C_n(a,b) \overset{\phi}{=} C_p(a,b) \times C_q(a,b)$, for any two points $P, Q \in C_n(a,b)$,

$$P \oplus Q = \phi^{-1}(P_p \oplus Q_p, P_q \oplus Q_q). \quad (8)$$

$(C_n(a,b), \oplus)$ constructs a finite Abel group in [10], where $\oplus$ is defined as equation (8).

Theorem 1: Let $A \in C_n(a,b)$, the order of A is the minimal positive integral k such that kA=0, and denote $o(A) = k$. $\forall A = (x,y) \in C_n(a,b)$,there is a unique point A in $C_p(a,b) \times C_q(a,b)$ response to the point (A_p, A_q) and the order of A

$$o(A) = lcm(o(A_p), o(A_q)).$$

Corollary: let p, q two distinctness large prime and n=pq, such that $(\frac{a}{p})=(\frac{a}{q})=-1$, and $p+1 = 2r$, $q+1=2s$, where both r and s are prime, then there exist one point G in the curve $C_n(a,b)$, which order $N_n = 2rs$.

The above proof can be found in [10].

Theorem 2: Let conic curve $C_n(a,b)$, where $n = pq$ (p,q : prime). Let $N_n = lcm(\#C_p(a,b), \#C_q(a,b))$, then for any $P \in C_n(a,b)$ and any integer k , we have:

$$(k \cdot N_n + 1) \cdot P \equiv P \pmod n.$$

Proof: By the above Theorem, for any $P \in C_n(a,b)$, there exist a unique point corresponding (P_p, P_q) in $C_p(a,b) \times C_q(a,b)$, and $O(P)$=lcm($O(P_p)$,$O(P_q)$), clear $O(P) | N_n$, so we have shown the above identity。

4 Conic Curve Digital Signature Algorithm

We next describe the conic curve digital signature algorithm.

- Key Generation

Each entity A does the following:

1. Select a conic curve C defined over F_p. The number of points in C (F_p) should be divisible by a large prime n.

2. Select a point $P \in C$ (F_p) of order n.

3. Select a statistically unique and unpredictable integer d in the interval [1, n -1].

4. Compute $Q = dP$.

A's public key is (C , P, n, Q); A's private key is d.

- Signature Generation

To sign a message m, A does the following:

1. Select a statistically unique and unpredictable integer k in the interval [1, n - 1].

2. Compute kP = (x , y) and $r = x \bmod n$. (Here x is regarded as an integer, for example by conversion from its binary representation.)

If r = 0, then go to step 1. (This is a security condition: if r = 0, then the signing equation $s = k^{-1}\{h(m) + dr\} \bmod n$ does not involve the private key d!)

3. Compute $k^{-1} \bmod n$.

4. Compute $s = k^{-1} \{ h(m) + dr \} \bmod n$, where h is the Secure Hash Algorithm(*SHA-1*).

If s = 0, then go to step 1. (If s = 0, then $s^{-1} \bmod n$ does not exist; s^{-1} is required in step 2 of signature verification.)

The signature for the message m *is* the pair of integers (r, s).

- Signature Verification

To verify A's signature (r, s) on m, B should do the following :

1. Obtain an authentic copy of A's public key (C , P, n, Q). Verify that r and s are integers in the interval [1, n - 1]..

2. Compute $w = s^{-1} \bmod n$ and $h(m)$.

3. Compute $u_1 = h(m)w \bmod n$ and $u_2 = rw \bmod n$.

4. Compute $u_1P + u_2Q = (x_0, y_0)$ and $v = x_0 \bmod n$.

5. Accept the signature if and only if $v = r$.

Instead of each entity generating its own conic curve, the entities may elect to use the same curve C and point P of order n. In this case, an entity's public key consists only of the point Q. This results in

public keys of smaller sizes. Additionally, there are point compression techniques whereby the point $Q = (x_Q , y_Q)$ can be efficiently constructed from its x-coordinate x_Q and a specific bit of the y-coordinate y_Q.

5 Threshold Digital Signature

In this section, we present a new *(t, n)* threshold signature scheme with the assistance of a mutually trusted center. The scheme consists of three phases: the system initiation phase, the threshold signature generation phase, and the threshold signature verification phase. We describe the three phases in details as follows:

5.1 System Initialization Phase

The system contains a mutually trusted center, who is responsible for selecting all parameters. Assume that there are n members in a group, let A be the set of all group members. Any t or more members in the group can sign a message on behalf of the group, let B be any subset in A of size t . The mutually trusted center selects the following parameters: a number $N=pq=(2p'+1)(2q'+1)$, where $p,q,$ p' and q' are distinct large primes. A generator g with order $v= p'q'$ in Z_n^*. A system public value e such that $gcd(e, v)=1$, where $e \cdot d=1 \bmod v$ and d is a system secret value. A one-way hash function h() . A secret polynomial function $f(x)=f(x)=c_{t-1}x^{t-1}+ \cdots +c_1x+c_0 \bmod v$ with degree t-1, where $c_{n-1},\cdots,c_1,c_0 \in Z_v^*$. A secret key x and a public key y, where $x=f(0) \bmod v$ and $y=g^x \bmod N$.

Thus, the mutually trusted center publishes e,y,N,g and h( ), and keeps $d,x,v,p,q,$ p' and q' in secret.

For each group member U_i with a public value ID_i , for $i \in A$, the mutually trusted center computes U_i's secret key $x_i = (g^{f(ID_i)\cdot \ell_i})^d \bmod N$ and publishes his public key $y_i = g^{f(ID_i)\cdot \ell_i} \bmod N$, where $\ell_i = \prod_{j\in A, j\neq i} (ID_i - ID_j)^{-1} \bmod v$.

5.2 Threshold Signature Generation Phase

Without loss of generality, assume that there are t group members want to sign a message m on behalf of the group. The t group members can be denoted as $U_1,U_2, \cdots\cdots U_t$. The set of group members is denoted as B . Each member U_i chooses a random number k_i and computes r_i as

$$r_i = g^{k_i \cdot e} \bmod N$$

Thus, U_i makes r_i publicly available through a broadcast channel. After all r_i are available, each group member computes the product R as

$$R = \prod_{i\in B} r_i \bmod N$$

Then, U_i uses his secret key x_i and the random number k_i to compute

$$s_i = (x_i)^{h(m,R)\cdot \prod_{j\in A, j\neq B} (ID_i - ID_j)\cdot \prod_{j\in B, j\neq i} (0-ID_j)} \cdot g^{k_i} \bmod N$$

The user U_i sends $\{r_i, s_i\}$ to a designated clerk, who takes the responsibility of collecting the partial signatures. Besides, the clerk may authenticate the partial signatures by verifying the following equation

$$s_i^e = (y_i)^{h(m,R)\cdot \prod_{j\in A, j\neq B} (ID_i - ID_j)\cdot \prod_{j\in B, j\neq i} (0-ID_j)} \cdot r_i \bmod N$$

If the equation holds, the partial signature $\{r_i, s_i\}$ is valid.

Theorem 1: If

$$s_i^e = (y_i)^{h(m,R)\cdot \prod_{j\in A, j\neq B} (ID_i - ID_j)\cdot \prod_{j\in B, j\neq i} (0-ID_j)} \cdot r_i \bmod N$$

holds, then the partial signature $\{r_i, s_i\}$ is valid.

Further, the clerk computes the group signature S , where

$$S = \prod_{i\in B} s_i \bmod N$$

Thus, *(R, S)* is the group signature for the message m.

5.3 Threshold signature verification phase

Any verifier can use the group public key to authenticate the validity of the group signature *(R, S)* for the message m by checking the following equation

$$S^e \equiv y^{h(m,R)} \cdot R \bmod N$$

If the equation holds, the group signature *(R, S)* is valid.

Theorem 2: If $S^e \equiv y^{h(m,R)} \cdot R \bmod N$ holds, then the group signature $\{m, R, S\}$ is valid.

Proof: In the second phase, the individual signature $\{r_i, s_i\}$ of the message m satisfies the following equation

$$s_i^e = (y_i)^{h(m,R)\cdot \prod_{j\in A, j\neq B} (ID_i - ID_j)\cdot \prod_{j\in B, j\neq i} (0-ID_j)} \cdot r_i \bmod N$$

Since $R = \prod_{i \in B} r_i \bmod N$ and $S = \prod_{i \in B} s_i \bmod N$, we have

$$S^e \equiv (\prod_{i \in B} s_i)^e \bmod N$$
$$\equiv \prod_{i \in B} (y_i)^{h(m,R) \cdot \prod_{j \in A, j \neq B} (ID_i - ID_j) \cdot \prod_{j \in B, j \neq i} (0 - ID_j)} \cdot r_i \bmod N$$
$$\equiv \prod_{i \in B} (g^{f(ID_i) \cdot \prod_{j \in A, j \neq i} (ID_i - ID_j)^{-1} \; h(m,R) \cdot \prod_{j \in A, j \neq B} (ID_i - ID_j) \cdot \prod_{j \in B, j \neq i} (0 - ID_j)}) \cdot \prod_{i \in B} r_i \bmod N$$
$$\equiv \prod_{i \in B} (g^{f(ID_i) \cdot \prod_{j \in A, j \neq i} (ID_i - ID_j)^{-1} \; h(m,R) \cdot \prod_{j \in A, j \neq B} (ID_i - ID_j) \cdot \prod_{j \in B, j \neq i} (0 - ID_j)}) \cdot R \bmod N$$
$$\equiv \prod_{i \in B} (g^{h(m,R) \cdot f(ID_i) \cdot \prod_{j \in B, j \neq i} (ID_i - ID_j)^{-1} \cdot \prod_{j \in B, j \neq i} (0 - ID_j)}) \cdot R \bmod N$$

According to the reconstructed relation of the polynomial $f(x)$, the above equation can be rewritten as

$$g^{f(0) \cdot h(m,R)} \cdot R \bmod N$$
$$\equiv y^{h(m,R)} \cdot R \bmod N$$

Therefore, $S^e \equiv y^{h(m,R)} \cdot R \bmod N$ and the group signature $\{m, R, S\}$ can be verified.

As stated in the above Theorem, the verifier will believe that the group signature (R, S) for the message m is valid. However, it is not possible to find out the identities of the t signers from the group signature (R, S). Thus, the signers are anonymous.

5.4 Security Analysis

In the following, some possible attacks against the proposed scheme are presented. As we can see, none of these attacks can break our proposed scheme.

Attack 1: An adversary tries to reveal the group secret key $x = f(0) \bmod v$ from the public key y.

The adversary might directly solve x from the equation $y = g^x \bmod N$. This implies that the adversary can obtain x from the above equation if he solve the problem of computing a discrete logarithm modulo the composite number N.

Attack 2: Any t members of the group, may cooperate to reveal the secret keys $x = f(0) \bmod v$ and d.

Since each U_i is the group member, they have the corresponding secret key $x_i = g^{f(ID_i) \cdot d \cdot \ell_i} \bmod N$, where $\ell_i = \prod_{j \in A, j \neq i} (ID_i - ID_j)^{-1} \bmod v$. Thus, they generate a valid group signature. As for getting the secret keys, they might plot the following two approaches: (i) revealing the value $f(ID_i)$ from x_i or y_i, (ii) revealing the secret key d of the mutually trusted center. In the first approach, as analyzed in Attack 1, they will face the computation of the discrete logarithm modulo the composite number N. As for the other approach, since the modulus N is chosen to be infeasible to factor, specialized attacks applicable to the RSA scheme are ineffective, e.g., it is infeasible to find d with the known e. That is, our scheme can withstand the conspiracy attack.

Attack 3: The signer U_i colludes with the designated clerk that they may forge the valid signature for message m', but the signers $U_2, \cdots\cdots U_t$ reject to sign a message m' with him.

In Michels and Horster's forgery attack on Harn's schemes, the signer U_i wants to his victims, the signers $U_2, \cdots\cdots U_t$ to sign a message m' with him. They reject, but agree to sign the innocent message m with him. In our scheme, since the partial signature is

$$s_i = (x_i)^{h(m,R) \cdot \prod_{j \in A, j \neq B} (ID_i - ID_j) \cdot \prod_{j \in B, j \neq i} (0 - ID_j)} \cdot g^{k_i} \bmod N$$

and the message m is protected by the one-way hash function $f())$ with the value R. Thus, our scheme can withstand the forgery attack.

Attack 4: The designated clerk and an adversary may try to reveal the member's secret key x_i. Since the modulus N is chosen to be infeasible to factor, specialized attacks applicable to the RSA scheme are ineffective, e.g., it is infeasible to find d with the known e. They can not compute $x_i = (y_i)^d \bmod N$. This implies that the designated clerk and an adversary can obtain member's secret key x_i if they solve the factorization problem .

Attack 5: A receiver tries to determine the identities of the signers from the group signature.

A receiver only knows the group signature $\{R,S\}$ and may authenticate the validity. Since the receiver does not know the partial signatures $\{r_i, s_i\}$ of each signer U_i, he can not check $s_i^e = (y_i)^{h(m,R) \cdot \prod_{j \in A, j \neq B} (ID_i - ID_j) \cdot \prod_{j \in B, j \neq i} (0 - ID_j)} \cdot r_i \bmod N$

6 A Proxy Signature Scheme

The concept of the proxy signatures was introduced by Mambo *et al.* [11]. As the proxy signatures in

areas such as e-commerce and e-money has a good application prospects, it has triggered extensive research. Based on the above public key cryptography, we propose a Proxy Signature Scheme.

The conic curves equation and parameters are described above. It is assumed that a signer Alice asks a proxy signer Bob to carry out signing for her. (n_A, e_A) is the public key of original signer Alice, and her corresponding private key is (d_A, N_{n_A}), where $N_{n_A} = lcm(\#C_{p_A}(a,b), \#C_{q_A}(a,b))$. (n_B, e_B) is the public key of original signer Bob, and his corresponding private key is (d_B, N_{n_B}), where $N_{n_B} = lcm(\#C_{p_B}(a,b), \#C_{q_B}(a,b))$. e_p is a proxy public key. d_{p_A} is a proxy private key of Alice, and d_{p_A} is a proxy private key of Bob. Furthermore, a universal secure hash function h(•) should be published. The details are as follows.

6.1 System Initialization Phase

Alice carries out the steps in below:

(1) First make a warrant m_ω, which records the delegation policy including limits of authority, valid periods of delegation etc.

(2) Select a random number $e_p \in (1, \cdots, N_{n_A})$, and compute d_{p_A}, where

$$\gcd(e_p, N_{n_A}) = 1$$

$$e_p d_{p_A} \equiv 1 (\text{mod } N_{n_A})$$

(3) Calculate P_A and α, where

$$P_A = P(h(m_\omega)) = (x_A, y_A)$$

$$\alpha = P_A \cdot d_{P_A} \cdot d_A$$

(4) Send $(m_\omega, P_A, \alpha, e_p)$ to Bob.

6.2 Proxy Generation Phase

Bob first checks whether $e_p < N_{n_B}$ or $\gcd(e_p, N_{n_B}) = 1$. If it does not, he rejects those and stop.

Bob checks the equation $P_A = \alpha \cdot e_p \cdot e_A$ and $y_A x_A^{-1} \equiv h(m_\omega) (\text{mod } n_A)$. If it does not, Bob stop. Otherwise he compute d_{p_B}, where

$$e_p d_{p_B} \equiv 1 (\text{mod } N_{n_B}).$$

6.3 Signature Generation Phase

To sign a message m on behalf of Alice, Bob computes

$P_B = P(h(m)) = (x_B, y_B)$,

$\beta = P_B \cdot d_{P_B} \cdot d_B$.

Then $(\alpha, \beta, m_\omega, e_p, n_A, P_A, P_B, e_A, n_B, e_B)$ is a proxy signature of message m.

6.4 Verification Phase

Anyone can check whether (α , β , m_ω , e_p , n_A , P_A , P_B , e_A , n_B , e_B) is a valid proxy signature of message m by the follwing equation:

$P_A = \alpha \cdot e_p \cdot e_A$,

$y_A x_A^{-1} \equiv h(m_\omega) (\text{mod } n_A)$,

$P_B = \beta \cdot e_p \cdot e_B$,

$y_B x_B^{-1} \equiv h(m) (\text{mod } n_B)$.

If it holds, the signature will be accepted, otherwise rejected.

6.5 Security Discussion

We briefly discuss security of the proxy signature scheme we propose.

Unforgeability: Since β contains Bob's private key and proxy secret key, Only Bob can compute β to generate a valid proxy signature.

Veriflablity: Since α contains Alice's private key and proxy secret key, Bob can not compute α. Alice's agreement on m is also verified explicitly, because Alice's agreement has included in the proxy signature.

Identifiablity: From the verification equations(8), proxy signer Bob's public key information has been explicitly included in a valid proxy signature. Therefore, anyone can determine the identity of the corresponding proxy signer Bob.

Prevention of misuse: Due to using the proxy warrant, the proxy signer Bob can only sign messages that have been authorized by the original signer Alice.

7 A Public Key Cryptosystem Scheme on Conic Curves over the Ring Zn

In this section, we propose a public key cryptosystem scheme on conic curves over the ring Z_n. Let a, b $\in$ Z_n be two parameters. The conic

curves equation, denoted by $y^2 \equiv ax^2 - bx \pmod{n}$, satisfy the following condition:

(1) $(a,n) = (b,n) = 1$

(2) $n = pq$, where p and q are two large different primes.

(3) $(\frac{a}{p}) = (\frac{a}{q}) = -1$.

Key Generation: User U chooses large primes p and q. U computes the product $n = pq$, and $N_n = lcm(\#C_p(a,b), \#C_q(a,b)) = lcm(p+1, q+1)$. U chooses an integer e which is coprime to N_n, and computes an integer d such that

$$ed \equiv 1 \pmod{N_n}.$$

U's secret key is d and $(p, q, \#C_p(a,b), \#C_q(a,b), N_n)$. U's public key is (n, e).

Encryption: A plaintext $M = (m_x, m_y)$ is an integer pair, where $m_x \in Z_n$, $m_y \in Z_n$. Let $M = (m_x, m_y)$ be a point on the conic curve $C_n(a,b)$. Sender A encrypts the point M by encryption function E(•) with the receiver's public key e and n as

$$C = E(M) = e \cdot M,$$

and sends a ciphertext pair $C = (C_x, C_y)$ to a receiver B.

Decryption: Receiver B decrypts a point C by decryption function D(•) with his secret key d and public key n as

$$M = D(C) = d \cdot C.$$

Because $d \cdot C = d \cdot e \cdot M = (k \cdot N_n + 1) \cdot M = M$.

An addition operation on the points of an conic curve over the ring Zn can be defined that makes it into an abelian group. Compared with KMOV scheme on elliptic curves, our scheme is not need special conic curves over the ring Z_n to construct public key cryptosystem. This has enabled our scheme to have a more extensive application. In addition, some operations on the conic curves will be relatively easy, it has enabled our scheme to greatly enhance efficiency.

The conic curves equation and parameters are described above. Before signing a message m, a hashing function HASH() should be applied. HASH(m) embedded on $C_n(a,b)$ is a point M.

Alice release as public parameters n, a, b and e. Then she computes the point $Q = (s, t)$ on $C_n(a,b)$ according to

$$Q = (s,t) = d \cdot M.$$

The signature for the message m is the pair (s,t), which can be checked by computing

$$M = e \cdot Q$$

on $C_n(a,b)$ and extracting the message m from M (because $(ed) \cdot M = M$).

The security of our scheme over conic curves is based on the difficulty of factoring n. In this section, we discuss the security of these schemes from various viewpoints.

The original RSA schemes can be broken if one can determine order of the multiplicative groups. It is known that finding $\phi(n) = (p-1)(q-1)$ is computationally equivalent to factoring n. In our proposed schemes, a similar relationship holds.

Theorem 3: Let N_n be $lcm(p+1, q+1)$. Finding N_n is computationally equivalent to factoring the composite number n.

The security of the original RSA scheme is also based on the difficulty of finding the secret multiplier key d. We have the following relationship.

Theorem 4: Solving a secret key d from public keys e and n is computationally equivalent to factoring a composite number n.

The encryption-decryption functions E(•) and D(•) for our scheme are homomorphic for addition as $E(M_1 + M_2) = E(M_1) + E(M_2)$ and $D(M_1 + M_2) = D(M_1) + D(M_2)$, for any points M_1 and M_2 on the same conic curve. The probability that randomly chosen integer pairs M_1 and M_2 are on the same conic curve is as negligibly small. Thus, passive attacks using homomorphism seem to be ineffective against our scheme.

Consider an active attack(a chosen-plaintext attack) using homomorphism. Suppose an attacker A wants to make a victim B sign a plaintext $M = (m_x, m_y)$ without B's consent. A generates another message M' with B's public keys (e_B, n_B) and random integer r,

$$M' = M + e_B \cdot (r \cdot M),$$

and sends M' to B. B makes a signature S' for M' with his secret key d_B:

$$S' = d_B \cdot M' = d_B \cdot (M + e_B \cdot (r \cdot M)).$$

Then, A computes a signature S for M from S' by

$$S = S' - r \cdot M .$$

Using this technique, A can forge B's signatures without B's secret key. To counter this attack, a randomization of a plaintext with a hashing function should be applied.

Isomorphism Attacks are same as homomorphism attacks.

8 A Distribution Scheme Verification Based on Conic Curve

The following is a recognizable distribution scheme based on conic curve. This scheme can verify the people who sign the message and the conference keys. The security level is based on the difficulties of solving conic curve discrete logarithm problems[9].

Assume the system parameters are ($\boldsymbol{F}_p$, C , P, n). A and B are communicating using k_A, k_B for the private keys respectively, and the public keys are $P_A = K_AP$, $P_B = K_BP$. The sign equation is $s = t^{-1}(e+rk)$. This is private key distribution, so we can take $e = 1$, and the sign equation is $s = t^{-1}(1+rk)$.

Step 1: A random chooses an integert, with $1<t<n$, and a conference private key k, and then maps k to G with C ($\boldsymbol{F}_q$).The function of mapping is public:

1. A computes $Q=G+tP_B$, $tP = (x,y)$, take $r \equiv x \pmod n$
2. A computes $S \equiv t^{-1}(1+rk) \pmod n$, then A passes (Q, r, s) to B

Step 2: B receives (Q, r, s), then computes:

1. $s^{-1} \pmod n$
2. $M = (x^*, y^*) = s^{-1}(P + rP_A)$

If $x^* \equiv r \pmod n$, then B acceptes A's signature, say verify A. B then will computes: $G^* = Q - k_BM$, and maps G^* to k. This is the conference private key.

9 A Distributed E-Business Authentication Scheme

Let A stand for registration server, and B stand for clients. the public keys for A and B are P_A and P_B, and the private keys are k_A and k_B, respectively. Conic curve is $C(\boldsymbol{F}_q)$, the base point is P, order is n, and $P_A=k_AP$, $P_B=k_BP$.

The following is the process of registration:

Client B submit user ID to registration server A, and the process may be transmitted securely via SSL. The use ID could be users' names (or the place where users attending) and software's ID, or could even be some identification information, for example, CPU ID, the series number of hard disk, MAC address of the network adapters, etc. Then, registration server A will generate the registration code according to received ID, and sign it and send it back to client B. Client B will verify the license by the signature to determine whether it comes from server A. If it is, sever B will submit license to register and after sever A's verification, the registration succeeds and create an entry for client B in server A.

9.1 Generate the licence

B passes A the user ID, then A computes licence using the following steps:

1. Random choose an integer t ($1<t<n$), compute $R= tP = (x', y')$.
2. make the coordinates x' and y' of P_B and R as parameters, compute SHA value, say Hash = SHA (ID, x, y).
3. Compute $s_n \equiv t - \text{Hash}*k_A \pmod n$
4. Make s_n and Hash as license.

9.2 Generate the licence

Step 1: mapping lience to G on $C(\boldsymbol{F}_q)$ publicly:

1. A computes $Q = G + tP_B$, $tP=(x, y)$, $r \equiv x \pmod n$.
2. A computes $s = t^{-1}(1+rk_A) \pmod n$.

Then A passes (Q, r, s) to B.

Step 2: B receives (Q, r, s), then computes:

1. A computes $s^{-1} \pmod n$.
2. $M = (x^*, y^*) = s^{-1}(P + rPA)$.

If $x^* \equiv r \pmod n$, then B acceptes A's signature, say verify A. B then will compute: $G^* = Q - k_BM$, map G^* to k. his is the private key.

9.3 Registration and Verification

B submits license . If it can pass the verification, then A can register successfully.

1. Abstract s_n and Hash.
2. Compute $R \equiv s_n*P + Hash*P_A \pmod n = (x, y)$
3. Make the coordinates x' and y' of ID and R as parameters, compute SHA value, say $Hash = SHA$ (ID, x, y).

4. If H = Hash, the registration is successful; otherwise fails.

9.4 Analysis and Discuss

Now we study the securityt of our scheme that based on the securityt of conic curve digital signature algorithm . The basis for the security of conic curve cryptosystems such as conic curve digital signature algorithm is the apparent intractability of the following conic curve discrete logarithm problem : Given an conic curve C defined over $\boldsymbol{F}_q$, a point $P \in C(\boldsymbol{F}_q)$ of order N, and a point $Q \in C(\boldsymbol{F}_q)$, determine the integer x, $0 \leqq x \leqq N-1$, such that $Q = xP$, provided that such an integer exists.

1. In the process of generating and verifying registration codes, $C(\boldsymbol{F}_q)$, P, public key P_A, private key k_A are used. A registration machine to get k_A can only be made by obtaing public P_A, P and $P_A = k_A P$. However, this has to solve the conic curve discrete logarithm problem, which is very difficult.

2. If eavesdropper intercepts the ID from B to A in 4.1, and substitute it as him self's, A will generate registration code as usual without knowing this. Even though eavesdropper can intercepts (Q, r, s) in 4.2, due to computing G^* needs k_B, and solve k_B only with G^*, Q, M is solving conic curve discrete logarithm problem. Eavesdropper cannot get the registration code. Therefore, the registration codes distribution is important and necessary.

3. The derive of (1) in 4.2 is:

$M=(x^*, y^*)= s^{-1}(P+rP_A)= s^{-1}(1+rk_A)P$

Based on Step1's (2) in 4.2:

$s \equiv t^{-1}(1+rk_A) \bmod n$, so we have: $M = tP$

4. the derive of (2) in 4.2 is:

$G^*=Q - k_B M= G+tk_B P - k_B tP= G$

5. the derive of (3) in 4.2 is:

$sn^*P+Hash^*P_A=tP=R$.

10 Conclusion

E-businesses need some form of assurance of the security provided in the technology products they purchase. For such assurance, there are international standards used to validate vendors' security claims against established criteria informal evaluations. Security evaluations are carried out by independent, licensed and accredited organizations. The evaluation process, from inception to certificate, often lasts up to a full year (and sometimes longer). Vendors who have undergone evaluations of their products learn to improve upon their development, testing and shipping processes as a result of completing the demanding process. Security evaluations are perhaps the most effective way to qualify a vendor's assertions about its security implementations. Is a product that has not completed such evaluations secure enough to run an e-business? Is it secure enough to protect an organization's most sensitive data? E-businesses demand that the software and hardware vendors they select ship certified, provably-secure products. Assurance afforded by independent security evaluations lets e-businesses be assured of the products they purchase and deploy.

In the real world, a company's credibility and large sums of money are atstake. How can businesses protect themselves against liability if a cryptosystem that was supposed to be secure is broken and thousands of credit card numbers are stolen? The answer is that various standards bodies with professional organizations such as ANSI, IEEE, and ISO, evaluate and make recommendations for the use of approved cryptosystems. If companies use products that adhere to these guidelines, then they are largely protected from any possible lawsuit if a system is broken. It would be extremely risky for a company to sell a product with a type of cryptography that has not been approved by the major standards bodies. Standards bodies typically include representatives of various constituencies and professional groups, not all of whom are knowledgeable about the mathematics of cryptography. Before a cryptosystem is included in the recommendations of a standards body, a large number of people have the opportunity to raise objections either to the cryptosystem in its entirety or to the proposer's suggestions for implementation (choice of parameters, methods of generating keys, etc.). Naturally, marketers of competing cryptosystems have a strong incentive to something wrong with the new system. And no one wants to end up in the embarrassing situation of having approved a system that is broken a few months later. So it is not surprising that standards bodies tend to be conservative and slow-moving. In the case of the most popular current public-key cryptosystems, the time lags between academic publication of a proposal for a type of cryptography and approval of specic recommendations for its practical use were roughly 15 years.

As technology matures and secure e-business systems are deployed, companies will be better positioned to manage the risks associated with

disintermediation of data access. Through this process businesses will enhance their competitive edge while also working to protect critical business infrastructures from malefactors like hackers, disgruntled employees, criminals and corporate spies.

In this paper, we propose a distributed electronic authentication scheme based on conic curves. This scheme is composed of two parts, constructing license and validating license. Because the security of license is determined by private key, not the arithmetic itself, user can not construct new license by given license and the public key as long as the private key is not leaked. Since encoding and decoding over conic are easily implement on conic curves, it has enabled our scheme to greatly enhance efficiency. We also analysis its security. The entire process guarantees the security and reliability.

References:

[1] M. Zhang, Factoring integers with conics, *Journal of Sichuan University (Natural Science)(in Chinese)*, Vol.33, No.4, 1996, pp. 356-359.

[2] Z. Cao, A public key cryptosystem based on a conic over finite fields Fp, *Advances in Cryptology: Chinacrypt98, Science Press (in Chinese)*, 1998, pp.45-49.

[3] Z. Cao. Conic analog of RSA cryptosystem and some improved RSA cryptosystems. *Journal of Naturul Science of Heilongjiang University (in Chinese)*, Vol.16, No.4, 1999, pp. 15-18.

[4] Z. Dai, D. Pei, J. Yang. et al, Cryptanalysis of a public-key cryptosystem based on conic curves. *CrypTECc99 (HongKong)*, 1999.

[5] Y. Desmedt and Y. Frankel, "Threshold Cryptosystem", *In Proc. of Crypto'89*, Lecture Notes in Computer Science, LNCS 435, Springer Verlag, 1990, pp.307-315.

[6] T. Hwang, "Cryptosystem for group oriented cryptography", *In Proc. of Eurocrypt'90*, Lecture Notes in Computer Science, LNCS 473, Springer Verlag, 1991, pp.352-360.

[7] T.P. Pedersen, "Distributed Provers with Applications to Undeniable Signatures", *In Proc. of Eurocrypt'91*, Lecture Notes in Computer Science, LNCS 547, Springer Verlag, 1991, pp.221-238.

[8] Feldman, "A Practical Scheme for Non-Interactive Verifiable Secret Sharing", *In Proc. of 28th IEEE symposium on Foundations of Computer Science*, 1987, pp.427-437.

[9] Y. Desmedt and Y. Frankel, "Shared Generation of Authenticators and Signatures", *In Proc. of Crypto'91, Lecture Notes in Computer Science*, LNCS 576, Springer Verlag, 1991, pp.457-469.

[10] C. Park and K. Kurosawa, "New ElGamal Type Threshold Digital Signature Scheme", *IEICE Trans*. Fundamentals, E79-A(1):86-93, January 1996.

.