New Public-Key Schemes Based on Elliptic Curves over the Ring $\mathbf{Z}_n$

Kenji Koyama*
Ueli M. Maurer†
Tatsuaki Okamoto‡
Scott A. Vanstone§

ABSTRACT Three new trapdoor one-way functions are proposed that are based on elliptic curves over the ring $\mathbf{Z}_n$. The first class of functions is a naive construction, which can be used only in a digital signature scheme, and not in a public-key cryptosystem. The second, preferred class of function, does not suffer from this problem and can be used for the same applications as the RSA trapdoor one-way function, including zero-knowledge identification protocols. The third class of functions has similar properties to the Rabin trapdoor one-way functions. Although the security of these proposed schemes is based on the difficulty of factoring n, like the RSA and Rabin schemes, these schemes seem to be more secure than those schemes from the viewpoint of attacks without factoring such as low multiplier attacks.

1 Introduction

In their seminal 1976 paper [3], Diffie and Hellman introduced the concept of a trapdoor one-way function (TOF). A TOF is a function that is easy to evaluate but infeasible to invert, unless a secret trapdoor is known, in which case the inversion is also easy. Although no realisation of a TOF was proposed in [3], Diffie and Hellman observed that such a function would allow the construction of digital signature schemes and public-key cryptosystems, two concepts that they introduced.

The first implementation of a TOF was proposed by Rivest, Shamir and Adleman in 1978 [21]. Its security relies on the difficulty of factoring a composite number n. Some other implementations [20, 4] of TOFs have been proposed based on the difficulty of factoring and discrete logarithms. ¿From another direction, one of the recent topics in the field of elliptic curves is their applicability to cryptography. The points of an elliptic curve E over a *finite field* form an abelian group. Hence the group E can be used to implement analogs of the Diffie-Hellman key exchange scheme and the ElGamal public key cryptosystem, as explained in [9]. The security of these analogous systems rests on the difficulty of the discrete logarithm problem on an elliptic curve.

In this paper, we propose new TOFs (or public-key cryptographic schemes) based on elliptic curves over a *ring* $\mathbf{Z}_n$, although an elliptic curve E over $\mathbf{Z}_n$ does not form a group.

*NTT Laboratories, Sanpeidani, Inuidani, Seikacho, Kyoto, 619-02, Japan
†Princeton University, Princeton, NJ 08544, USA; Supported by Omnisec AG, Switzerland
‡NTT Laboratories, Yokosuka-shi, Kanagawa 238-03, Japan
§University of Waterloo, Ontario, N2L 3G1, Canada

The security of these TOFs depends on the difficulty of factoring n. Although these schemes are less efficient than the RSA and Rabin schemes, our schemes seem to be more secure from the viewpoint of some attacks that do not use factoring such as low multiplier attacks. ¿From the same reason, even when the RSA system can be broken without factoring the modulus, our schemes seem to remain secure.

We begin with a brief review of the basic definitions and facts about elliptic curves over a finite field in Section 2. In Section 3, we show some properties of elliptic curves over a ring, which are used in the succeeding sections. Section 4 proposes a naive construction of the TOF (Type 0 scheme) which is based on elliptic curves over a ring, which can be used only in a digital signature scheme, and not in a public-key cryptosystem. In Section 5, we propose the Type 1 scheme which is based on the elliptic curve over a ring, and discuss its properties. In Section 6, we propose the Type 2 scheme based on the elliptic curve over a ring, and discuss its properties. Section 7 discusses the security of the proposed schemes, and Section 8 discusses their performance.

2 Elliptic Curves over a Finite Field

Let K be a field of characteristic $\neq 2, 3$, and let $a, b \in K$ be two parameters satisfying $4a^3 + 27b^2 \neq 0$. An elliptic curve over K with parameters a and b is defined as the set of points (x, y) with $x, y \in K$ satisfying the equation

$$y^2 = x^3 + ax + b,$$

together with a special element denoted $\mathcal{O}$ and called the point at infinity. We will mainly be interested in elliptic curves over the finite field $\mathbf{F}_p$ with p elements, for some prime p. Such a curve will be denoted $E_p(a, b)$. What makes elliptic curves interesting in cryptography is the fact that an addition operation on the points of an elliptic curve can be defined that makes it into an abelian group. This addition operation, which has but its name in common with the ordinary addition of integers, is described in the following.

Let E be an elliptic curve, and let P and Q be two points on E. The point $P + Q$ is defined according to the following rules. If $P = \mathcal{O}$, then $-P = \mathcal{O}$, and $P + Q = Q$ (i.e., $\mathcal{O}$ is the neutral element of E). Let $P = (x_1, y_1)$ and $Q = (x_2, y_2)$. If $x_1 = x_2$ and $y_1 = -y_2$, then $P + Q = \mathcal{O}$ (i.e., the negative of the point (x, y) is the point $(x, -y)$). In all other cases the coordinates of $P + Q = (x_3, y_3)$ are computed as follows. Let λ be defined as

$$\lambda = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1} & \text{if } x_1 \neq x_2 \\ \dfrac{3x_1^2 + a}{2y_1} & \text{if } x_1 = x_2. \end{cases}$$

(When $P + Q \neq \mathcal{O}$, then the denominator is always non-zero and thus the quotient is defined.) The resulting point $P + Q = (x_3, y_3)$ is defined by

$$\begin{aligned} x_3 &= \lambda^2 - x_1 - x_2 \\ y_3 &= \lambda(x_1 - x_3) - y_1. \end{aligned}$$

Clearly, the first equation is equivalent to $x_3 = \lambda^2 - 2x_1$ when $P = Q$. All computations are in the field over which E is defined. In particular, when the field is $\mathbf{F}_p$, all computations are modulo p. It is straightforward to verify that the defined addition operation satisfies the axioms for a group, i.e., that E is closed under addition and that addition is commutative and associative. (The existence of a neutral element and of inverse elements was mentioned above.)

Let $\#E_p(a,b)$ denote the order (i.e., the number of points) of the elliptic curve $E_p(a,b)$. It is well-known that $\#E_p(a,b) = p+1+t$ where $|t| \le 2\sqrt{p}$ for every elliptic curve over $\mathbf{F}_p$. Every value of t within the given bounds is taken for some pair (a,b), but this fact will not be used in this paper. There exists a polynomial-time algorithm due to Schoof [22] for computing the order of an elliptic curve, but this algorithm is quite impractical for large p. It is known that $E_p(a,b)$ is either cyclic or the product of two cyclic groups. In the latter case, $E_p(a,b) \cong \mathbf{Z}_{N_1} \times \mathbf{Z}_{N_2}$ where $N_1 \cdot N_2 = \#E_p(a,b)$, where N_2 divides N_1 and where N_2 also divides $p-1$. We refer to [9] for a more detailed introduction to elliptic curves, and to [8] for some further cryptographically useful properties of elliptic curves.

If the forms of elliptic curve $E_p(a,b)$ and prime p are restricted, the order $\#E_p(a,b)$ and the group structure are known as follows.

Lemma 1. *Let p be an odd prime satisfying $p \equiv 2 \pmod 3$. Then for $0 < b < p$ $E_p(0,b)$ is a cyclic group of order*

$$\#E_p(0,b) \;=\; p+1.$$

Lemma 2. *Let p be a prime satisfying $p \equiv 3 \pmod 4$. Then for $0 < a < p$ we have*

$$\#E_p(a,0) \;=\; p+1.$$

Moreover, $E_p(a,0)$ is cyclic if a is a quadratic residue modulo p and else $E_p(a,0) \cong \mathbf{Z}_{(p+1)/2} \times \mathbf{Z}_2$.

3 Elliptic Curves over a Ring

We now consider elliptic curves over the ring $\mathbf{Z}_n$, where n is an odd composite squarefree integer. (An alternative notation for $\mathbf{Z}_n$ used in the literature is $\mathbf{Z}/n\mathbf{Z}$.) Similar to the definition of $E_p(a,b)$, an elliptic curve $E_n(a,b)$ can be defined as the set of pairs $(x,y) \in \mathbf{Z}_n^2$ satisfying $y^2 \equiv x^3 + ax + b \pmod n$, together with a point $\mathcal{O}$ at infinity. An addition operation on $E_n(a,b)$ can be defined in the same way as the addition operation on $E_p(a,b)$, simply by replacing computations in $\mathbf{F}_p$ by computations in $\mathbf{Z}_n$. However, two problems occur. The first problem is that because the computation of λ requires a division which in a ring is defined only when the divisor is a unit, the addition operation on $E_n(a,b)$ is not always defined. The second problem, which is related to the first is that $E_n(a,b)$ is not a group. It seems therefore impossible to base a cryptographic system on $E_n(a,b)$. In the following we present a natural solution to these problems.

For the sake of simplicity, let $n = pq$ in the sequel be the product of only two primes as in the RSA system. Moreover, the addition operation on $E_n(a,b)$ described above, whenever it is defined, is equivalent to the (componentwise defined) group operation on $E_p(a,b) \times E_q(a,b)$. By the Chinese Remainder Theorem, every element c of $\mathbf{Z}_n$ can be represented uniquely as a pair $[c_p, c_q]$ where $c_p \in \mathbf{Z}_p$ and $c_q \in \mathbf{Z}_q$. Thus every point $P = (x,y)$ on $E_n(a,b)$ can be represented uniquely as a pair $[P_p, P_q] = [(x_p, y_p), (x_q, y_q)]$ where $P_p \in E_p(a,b)$ and $P_q \in E_q(a,b)$, with the convention that $\mathcal{O}$ is represented by $[\mathcal{O}_p, \mathcal{O}_q]$, where $\mathcal{O}_p$ and $\mathcal{O}_q$ are the points at infinity on $E_p(a,b)$ and $E_q(a,b)$, respectively. By this mapping, all elements of $E_p(a,b) \times E_q(a,b)$ are exhausted except the pairs of points $[P_p, P_q]$ for which exactly one of the points P_p and P_q is the point at infinity. Note that the addition operation on $E_n(a,b)$ described above is undefined if and only if the resulting point, when interpreted as an element of $E_p(a,b) \times E_q(a,b)$, is one of these special points.

It is important to note that when all prime factors of n are large, it is extremely unlikely that the sum of two points on $E_n(a,b)$ is undefined. In fact, if the probability of the addition operation being undefined were non-negligible, then the very execution of a computation on $E_n(a,b)$ would be a feasible factoring algorithm, which is assumed not to exist. Therefore, the first problem can be solved by considering the occurence probability.

The second problem, that $E_n(a,b)$ is not a group, can be solved by the following lemma. That is, although we cannot use the properties of a finite group directly, we can use a property of $E_n(a,b)$ which is similar to that of a finite group. The following lemma can be easily obtained from the Chinese Remainder Theorem.

Lemma 3. *Let $E_n(a,b)$ be an elliptic curve such that $\gcd(4a^3+27b^2,n)=1$ and $n=pq$ (p,q: prime). Let N_n be $\mathrm{lcm}(\#E_p(a,b),\#E_q(a,b))$. Then, for any $P \in E_n(a,b)$, and any integer k,*

$$(k \cdot N_n + 1) \cdot P = P.$$

4 Naive Construction of TOF Based on Elliptic Curves over a Ring

In this section, we show a naive construction of TOFs (Type 0 scheme) which are based on elliptic curves over a ring. These TOFs can be used only in a digital signature scheme, and not in a public-key cryptosystem. The flaws of the TOFs of this section are elliminated in the Type 1 and 2 schemes shown in following sections.

A digital signature scheme based on $E_n(a,b)$ can be set up as follows. The signer Alice chooses two primes p and q (or, more generally, a set of two or more distinct primes) and two parameters a and b satisfying $\gcd(4a^3+27b^2,n)=1$, where $n=pq$. She then computes the orders of the elliptic curves $E_p(a,b)$ and $E_q(a,b)$ (for example using Schoof's algorithm [22]), chooses a public encryption multiple e relatively prime to both $\#E_p(a,b)$ and $\#E_q(a,b)$, and computes the secret decryption multiple d according to

$$d \equiv e^{-1} \pmod{\mathrm{lcm}(\#E_p(a,b),\#E_q(a,b))}.$$

Alice releases as public parameters n,a,b and e. When she later wants to sign a message M she associates a point $P=(x,y) \in E_n(a,b)$ with M in a publicly-known way (see below) and computes the point $Q=(s,t)$ on $E_n(a,b)$ according to

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

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

$$P=(x,y)=e \cdot Q$$

on $E_n(a,b)$ and extracting the message M from (x,y) (because $(ed) \cdot P = P$ from Lemma 3).

Here, given a message M, a point (x,y) on $E_n(a,b)$ can efficiently be associated with M. M is first padded with sufficient redundancy, for instance by appending zero's to M, resulting in M'. x is defined as the smallest integer greater or equal to M' such that x^3+ax+b is a quadratic residue modulo n, and y is defined as one of the square roots modulo n of this number.

The flaws of this scheme are as follows:

(1) Schoof's algorithm [22] to compute $E_p(a,b)$ and $E_q(a,b)$ is infeasible for large p.
(2) The signature is roughly twice as long as the original message M.
(3) This scheme cannot be used for a public-key cryptosystem, since knowledge of the trapdoor is required to create a point on $E_n(a,b)$, which corresponds to a plaintext.

5 Basic TOF Based on Elliptic Curves over a Ring

In this section, we propose a new TOF (Type 1 scheme) that is based on elliptic curves over a ring. It overcomes the three flaws of the Type 0 scheme. For simplicity, we show a protocol for a public-key cryptosystem in the case of Lemma 1. We can easily construct a public-key cryptosystem in the case of Lemma 2, and digital signature schemes, although we omit a description.

Step 0 (Key Generation) User U chooses large primes p and q such that

$$p \equiv q \equiv 2 \pmod 3.$$

U computes the product $n = pq$, and $N_n = \mathrm{lcm}(\#E_p(0,b), \#E_q(0,b)) = \mathrm{lcm}(p+1, q+1)$.

U chooses an integer e which is coprime to N_n, and computes an integer d such that

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

Summarizing, U's secret key is d, $(p,\ q,\ \#E_p(0,b),\ \#E_q(0,b),\ N_n)$, and U's public key is $n,\ e$.

Step 1 (Encryption) A plaintext $M = (m_x, m_y)$ is an integer pair, where $m_x \in \mathbf{Z}_n$, $m_y \in \mathbf{Z}_n$. Let $M = (m_x, m_y)$ be a point on the elliptic curve $E_n(0,b)$.

Sender A encrypts the point M by encryption function $\mathbf{E}(\cdot)$ with the receiver's public key e and n as

$$C = \mathbf{E}(M) = e \cdot M \text{ over } E_n(0,b),$$

and sends a ciphertext pair $C = (c_x, c_y)$ to a receiver B.

Step 2 (Decryption) Receiver B decrypts a point C by decryption function $\mathbf{D}(\cdot)$ with his secret key d and public key n as

$$M = \mathbf{D}(C) = d \cdot C \text{ over } E_n(0,b).$$

[Notes]

1. In the case of Lemma 1, the minimum possible value of e is 5 because $2|N_n$ and $3|N_n$. In the case of Lemma 2, the minimum possible value of e is 3 because $2|N_n$.

2. For elliptic curves, the addition formula is independent of a and b, and the doubling formula is independent of b. Thus, the above protocol does not require computation of the value $b = y^2 - x^3 \bmod n$. If Lemma 2 is adopted, for the addition formula the sender S must compute a such that $a = (m_y^2 - m_x^3)/m_x \bmod n$, and the receiver R must compute a such that $a = (c_y^2 - c_x^3)/c_x \bmod n$.

6 Rabin-type Generalization

6.1 Protocol

We propose another TOF (Type 2 Scheme) also based on elliptic curves over a ring, which is the Rabin-type generalization of the basic TOF (Type 1 scheme). Type 2 scheme also overcomes the three flaws of the Type 0 scheme. For simplicity, we also show a protocol for a public-key cryptosystem in the case of Lemma 1.

Step 0 (Key Generation) User U chooses large primes p and q such that

$$p \equiv q \equiv 2 \pmod 3.$$

U computes the product $n = pq$, and the orders $N_p = \#E_p(0, b) = p + 1$ and $N_q = \#E_q(0, b) = q + 1$.

Summarizing, U's secret key is $p,\ q,\ N_p,\ N_q$, and U's public key is n.

Step 1 (Encryption) A plaintext $M = (m_x, m_y)$ is an integer pair, where $m_x \in \mathbf{Z}_n$, $m_y \in \mathbf{Z}_n$. Let $M = (m_x, m_y)$ be a point on the elliptic curve $E_n(0, b)$.

Sender A encrypts the point M by doubling on the elliptic curve E_n with the receiver's public key n as

$$C = 2 \cdot M \text{ over } E_n(0, b),$$

and sends a ciphertext pair $C = (c_x, c_y)$ to a receiver B.

Step 2 (Decryption) Receiver B computes $M_p \in E_p(0, b)$ and $E_q(0, b) \in E_q(0, b)$ from $C_p = (c_x \bmod p,\ c_y \bmod p) \in E_p(0, b)$ and $C_q = (c_x \bmod q,\ c_y \bmod q) \in E_q(0, b)$ such that

$$C_p = 2 \cdot M_p \text{ over } E_p(0, b), \quad C_q = 2 \cdot M_q \text{ over } E_q(0, b),$$

by using a halving algorithm, which is described in Section 6.2.

B computes $M = (m_x, m_y) \in E_n$ from $M_p = (m_{px}, m_{py}) \in E_p(0, b)$ and $M_q = (m_{qx}, m_{qy}) \in E_q(0, b)$ using the Chinese Remainder Theorem.

[Notes]

1. Since both N_p and N_q are even, 2 is not coprime to N_p, N_q and N_n.
2. Type 2 scheme has the drawback that there is 4:1 ambiguity in the decrypted messages, as is true for the original Rabin scheme.
3. In decryption based on a halving formula, the algorithm for finding a non-double point requires an exact expression of the elliptic curve. Thus, the receiver B must compute b such that $b = c_y^2 - c_x^3 \bmod n$.

6.2 Halving Algorithm

In general, points on $E_p(a,b):\ y^2 = x^3 + ax + b \bmod p$ can be separated into 2 classes, as integers in $\mathbf{Z}_p$ are classified into quadratic residue and quadratic non-residue modulo p.

Definition If $P = 2 \cdot X$ over $E_p(a,b)$ for some point X on the curve $E_p(a,b)$, then we call point P a *double point*, denoted by $P \in DP_p$. If $P \neq 2 \cdot X$ over $E_p(a,b)$ for any point X, then we call point P a *non-double point*, denoted by $P \in NDP_p$. □

Double point and non-double point are distinguishable by using the following three lemmas, when the group structure of $E_n(a,b)$ is known.

Lemma 4. Assume that E' be a cyclic subgroup of $E_p(a,b)$ with the maximum order of N'. Let P be in E', and N' be even. Then

$$P \in DP_p \text{ if and only if } N'/2 \cdot P = \mathcal{O} \text{ over } E_p(a,b),$$

Lemma 5. Assume that E' be a cyclic subgroup of $E_p(a,b)$ with the maximum order of N'. Let α be the number of the points $\in DP_p$ in E'. Then

$$\alpha = \begin{cases} N'/2, & \text{if } N' \text{ is even;} \\ N', & \text{if } N' \text{ is odd.} \end{cases}$$

Lemma 6. Assume that $E_p(a,b)$ has the group structure $\mathbf{Z}_{(p+1)/2} \times \mathbf{Z}_2$. Let E' be a cyclic subgroup of $E_p(a,b)$ with the maximum order of $(p+1)/2$ which includes point Q. Then

$$P \in DP_p \ \wedge \ P \in E' \text{ if and only if } e_{(p+1)/2}(P,Q) = 1 \ \wedge \ (p+1)/4 \cdot P = \mathcal{O} \text{ over } E_p(a,b),$$

where $e_{(p+1)/2}$ is the Weil pairing function [8, 19]. Note that $(p+1)/2$ is always even from the property of the finite Abelian group.

Next, consider a halving algorithm on elliptic curve $E_p(a,b)$ which outputs a half point of a given point over $E_p(a,b)$.

A halving algorithm on $E_p(a,b)$ can be constructed based on the Adleman-Manders-Miller algorithm [1, 11] for computing a square root mod p. Thus, we have the following theorem.

Theorem 7. There exists an expected polynomial time algorithm which, given an odd prime p, an elliptic curve $E_p(a,b)$ in the case of Lemma 1 or 2, N_p, and a point $Q \in DP_p$ as inputs, will output a half point of Q over $E_p(a,b)$.

The proof of Theorem 7 can be described explicitly in the following algorithm based on the Adleman-Manders-Miller algorithm.

Halving Algorithm on Elliptic Curve for Type 2 scheme

Input: p (prime), $E_p(a,b)$, N_p, $Q\ (= 2 \cdot H) \in E_p(a,b)$.

Step 1. Compute an odd c, and h such that $N_p = 2^h c$.

Step 2. Choose random point T such that $T \in NDP_p$ and T is in the maximum cyclic subgroup including Q.

Step 3. Set $Y = Q$, $H = (c+1)/2 \cdot Q$ over $E_p(a,b)$.

Step 4. Find the least k such that $(2^k c) \cdot Y = \mathcal{O}$ over $E_p(a,b)$.

Step 5. **If** $k = 0$ **then** output H; **else** set

$$Y = Y - 2^{h-k} \cdot T \text{ over } E_p(a,b), \quad H = H - 2^{h-k-1} \cdot c \cdot T \text{ over } E_p(a,b)$$

and **go to** step **4**.

Output: H.

An algorithm for finding a non-double point T is derived from Lemmas 4 and 6 as follows:

Algorithm 1 for Finding a Non-Double Point ($E_p(a, b)$: cyclic)

Input: p (prime), $E_p(a, b)$, N_p.

Step 1. Choose a random point $T = (t_x, t_y)$ on the curve.

Step 2. **If** T is a non-double point, that is, $N_p/2 \cdot T \neq \mathcal{O}$ over $E_p(a, b)$, **then** output T; **else go to** step **1**.

Output: $T = (t_x, t_y) \in NDP_p$.

Algorithm 2 for Finding a Non-Double Point ($E_p(a, 0) \cong \mathbf{Z}_{(p+1)/2} \times \mathbf{Z}_2$)

Input: p (prime), $E_p(a, 0)$, N_p, $Q \in E_p(a, 0)$.

Step 1. Choose a random point $T = (t_x, t_y)$ on the curve $E_p(a, 0)$ such that $e_n(Q, T) = 1$.

Step 2. **If** T is a non-double point, that is, $(p + 1)/4 \cdot T \neq \mathcal{O}$ over $E_p(a, 0)$, **then** output T; **else go to** step **1**.

Output: T such that $T = (t_x, t_y) \in NDP_p$, and $T \in E'$, where E' is a cyclic subgroup of $E_p(a, 0)$ with the maximum order of $(p + 1)/2$ which includes point Q.

There exists a polynomial time general algorithm for finding a point on the elliptic curve [9]. In case 1, for any $y \in \mathbf{Z}_p$, the point $((y^2 - b)^{1/3}, y)$ is on the curve. Since $3 \not| p - 1$, the value of $(y^2 - b)^{1/3}$ can be easily computed by $(y^2 - b)^{\beta} \bmod p$, where $3\beta \equiv 1 \pmod{p - 1}$. In case 2, for any $x \in \mathbf{Z}_p$, the point $(x, (x^3 + ax)^{1/2})$ is on the curve. Since $p = 4k + 3$ (k : integer), the value of $(x^3 + ax)^{1/2}$ can be easily computed by $(x^3 + ax)^{k+1} \bmod p$.

7 Security

The security of the proposed Type 1 scheme and Type 2 scheme over elliptic curves is based on the difficulty of factoring n. In this section, we discuss the security of these schemes from various viewpoints.

7.1 Solving the Order

In the original RSA and Rabin schemes in multiplicative groups, it is known that solving the order $\phi(n) = (p-1)(q-1)$ is computationally equivalent to factoring n. That is, the former is polynomially reducible to the latter, and vice versa. Similarly, in our proposed schemes (Types 1 and 2 in the cases of Lemmas 1 and 2), we have a similar relationship as follows.

Theorem 8. *Let N_n be $\mathrm{lcm}(\#E_p(a, b), \#E_q(a, b)) = \mathrm{lcm}(p + 1, q + 1)$. Solving N_n is computationally equivalent to factoring a composite number n.*

7.2 Solving the Secret Key

The security of the original RSA scheme is also based on the difficulty of solving the secret exponent key. The security of the Type 1 scheme is also based on the difficulty of solving the secret multiplier key d. We have the following similar relationship.

Theorem 9. *Let N_n be $\mathrm{lcm}(\#E_p(a, b), \#E_q(a, b)) = \mathrm{lcm}(p + 1, q + 1)$. Solving a secret key d from public keys e and n is computationally equivalent to factoring a composite number n.*

7.3 Complete Breakage

Completely breaking Type 1 and 2 schemes means to recover both m_x and m_y from any ciphertext pair (c_x, c_y) and the public keys. It is well known that completely breaking the original Rabin cryptosystem is as hard as factoring the composite n used as the modulus. For the Type 2 scheme, we have the following theorem.

Theorem 10. *Completely breaking Type 2 scheme is computationally equivalent to factoring n.*

Proof: It is clear that if once the factors of n are known, plaintext (m_x, m_y) can easily be computed from ciphertext (c_x, c_y) and public keys (a, n). Conversely, if there is an Algorithm A1, given P on $E_n(a,b)$ ($E_n(0,b)$ or $E_n(a,0)$), to output Q satisfying $P = 2 \cdot Q$ with non-negligible probablity, then we can construct an expected polynomial-time algorithm B to factor n, using A1 as an oracle. First, B chooses a random point $R = (r_x, r_y)$ ($r_x, r_y \in \mathbf{Z}_n$), and multiplies it by 2, asks A1 to halve this point, and B obtains R' satisfying $P = 2 \cdot R'$ with non-negligible probablity. Then B computes $R_0 = R - R'$. Since $2 \cdot R_0 = \mathcal{O}$, and R_0 over $E_p(a,b)$ (R_0 over $E_q(a,b)$) is $\mathcal{O}_p$ ($\mathcal{O}_q$), then R_0 is an undefined point with probability 1/2. If R_0 is undefined, B can compute a non-trivial factor of n by the extended Euclidean algorithm used for the division modulo n. Clearly, the expected running time of B is polynomial-time in $\log n$. □

In the Type 1 scheme, the equivalence between completely breaking this scheme and factoring n is *not* known. This situation is the same as the original RSA scheme.

7.4 Homomorphism Attacks and Their Countermeasures

The encryption-decryption functions $\mathbf{E}(\cdot)$ and $\mathbf{D}(\cdot)$ for Type 1 and 2 schemes are homomorphic for addition as

$$\mathbf{E}(M_1 + M_2) = \mathbf{E}(M_1) + \mathbf{E}(M_2) \ \text{ and } \ \mathbf{D}(M_1 + M_2) = \mathbf{D}(M_1) + \mathbf{D}(M_2),$$

for any points M_1 and M_2 on the *same* elliptic curve. This kind of homomorphic property is the basis for some attacking methods proposed [7] against the original RSA and Rabin schemes.

The probability that randomly chosen integer pairs M_1 and M_2 are on the same elliptic curve is as negligiblly small as $1/n$ for large number n. Thus, passive attacks using homomorphism seem to be ineffective against Type 1 and 2 schemes.

Consider an active attack (a chosen-plaintext attack) using homomorphism. Suppose an attacker A wants to make a victim B sign a plaintext $M = (m_x, m_y)$, which B may refuse to sign. A generates another message M' with B's public keys (e_B, n_B) and random integer r,

$$M' = M + e_B \cdot (r \cdot M) \text{ over } E_{n_B},$$

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

$$S' = d_B \cdot M' = d_B \cdot (M + e_B \cdot (r \cdot M)) \text{ over } E_{n_B}.$$

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

$$S = S' - r \cdot M \text{ over } E_{n_B}.$$

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 $\mathbf{h}$ should be applied before the application of the function $\mathbf{D}$, which is a technique similar to that for the original RSA-like schemes can be applied.

7.5 Isomorphism Attacks and Their Countermeasures

The following isomorphic property of the elliptic curves is known.

Lemma 11. Let E_{1n} and E_{2n} be elliptic curves such that

$$E_{1n}: \; y^2 = x^3 + a_1 x + b_1 \bmod n, \quad E_{2n}: \; y^2 = x^3 + a_2 x + b_2 \bmod n.$$

Let $M_1 = (m_{1x}, m_{1y})$, $C_1 = (c_{1x}, c_{1y}) \in E_{1n}$ and $M_2 = (m_{2x}, m_{2y})$, $C_2 = (c_{2x}, c_{2y}) \in E_{2n}$ where

$$C_1 = e \cdot M_1 \text{ over } E_{1n}, \quad C_2 = e \cdot M_2 \text{ over } E_{2n}.$$

Then the following statements are equivalent:

(i) E_{1n} and E_{2n} are isomorphic.

$$\text{(ii)} \; a_2 \equiv u^4 a_1 \bmod n, \quad b_2 \equiv u^6 b_1 \bmod n \quad \exists u \in \mathbf{Z}_n. \tag{1}$$

$$\text{(iii)} \; c_{2x} \equiv u^2 c_{1x} \bmod n, \quad c_{2y} \equiv u^3 c_{1y} \bmod n \quad \exists u \in \mathbf{Z}_n. \tag{2}$$

$$\text{(iv)} \; m_{2x} \equiv u^2 m_{1x} \bmod n, \quad m_{2y} \equiv u^3 m_{1y} \bmod n \quad \exists u \in \mathbf{Z}_n. \tag{3}$$

Proof: It is obvious from the proposition in [23] (pp.50-52) that two elliptic curves are isomorphic if and only if they have the same j-invariant. □

If C_1, C_2 and M_1 satisfying congruence (2) are given, then M_2 can be easily found by computing congruence (3). Notice that it is easy to check whether or not congruence (2) holds. If M_1 and M_2 are randomly chosen, then the probability that there exists u satisfying congruence (2) is a negligibly small $1/n$ for large n. Thus, passive attacks using isomorphism seem to be difficult for Types 1 and 2 schemes.

Consider an active attack (a chosen-plaintext attack) based on the isomorphic property of the elliptic curves. Suppose an attacker A wants to make a victim B sign a plaintext $M = (m_x, m_y)$, which B may refuse to sign. A generates another message M' with B's public key n_B and random integer u:

$$M' = (u^2 m_x \bmod n_B, \; u^3 m_y \bmod n_B),$$

and sends M' to B. B makes a signature $S' = (s'_x, \; s'_y)$ for M' with his secret key d_B:

$$S' = d_B \cdot M' \text{ over } E'_{n_B}.$$

Then, A computes a signature $S = (s_x, \; s_y)$ for M from S' by

$$S = (s_x, \; s_y) = (u^{-2} s'_x \bmod n_B, \; u^{-3} s'_y \bmod n_B).$$

Note that the curve E_{n_B} containing points $(M, \; S)$ and the curve E'_{n_B} containing points (M', S') are different, but isomorphic. Using this technique, A can forge B's signatures without B's secret key. To counter this attack, the same technique described in Section 7.4 can be applied.

An attacker may try to forge a signature by using both homomorphism and isomorphism shown above. However, such combined attacks can also be prevented by randomization with the hash function $\mathbf{h}$.

7.6 Security for Low Multiplier Attack

Hastad [6] showed a low exponent attack on the original RSA and Rabin schemes when the same message is encrypted with distinct plural moduli. He considered a problem of solving systems of congruences $P_i(m) \equiv 0 \pmod{n_i}$ $i = 1, ..., k$, where P_i are polynomial of degree e and the n_i are distinct relatively prime numbers and $m < \min n_i$. He proved that if $k > \frac{e(e+1)}{2}$, then m can be recovered in polynomial time. Thus, he pointed out that enciphering linearly related messages with the RSA scheme with low exponent or the Rabin scheme is insecure. For the original RSA scheme, let $c = m^e \bmod n$, $c_i = m_i^e \bmod n_i$, and $n = n_1 \cdot n_2 \cdots n_k$. In Hastad's algorithm, c is first obtained from c_i using the Chinese Remainder Theorem. Next, m is solved from a polynomial $c = m^e$ neglecting modulus n. For our proposed Types 1 and 2 schemes, let $C = e' \cdot M$ over $E_n(a, b)$, $C_i = e' \cdot M$ over E_{n_i}, where $C = (c_x, c_y)$, $M = (m_x, m_y)$, $C_i = (c_{ix}, c_{iy})$. The value of (c_x, c_y) is also obtained from (c_{ix}, c_{iy}). However, it is difficult to solve (m_x, m_y) from (c_x, c_y) because c_x and c_y are expressed by *rational* equations in m_x and m_y. Note that they cannot be expressed by polynomials. Since the rational equations include divisions modulo n, it seems impossible to compute by neglecting modulus n. Thus, even if the multiplier e' is small, a Hastad-like attack does not seem to work against the elliptic curve cryptosystems.

8 Performance

An elliptic curve addition $P_1 + P_2$ on $E_n(a, b)$ requires one division, one squaring operation and one general multiplication in $\mathbf{Z}_n$ when $P_1 \neq P_2$, and an extra squaring when $P_1 = P_2$. (The much faster additions and subtractions in $\mathbf{Z}_n$ are neglected for the sake of simplicity). Surprisingly, as opposed to $\mathbf{Z}_n$ where squaring can be performed faster than a general multiplication, doubling a point on an elliptic curve is computationally more costly than adding two different points. This means that in order to compute a multiple $c \cdot P$ of a point P, an irregular addition chain for c avoiding doubling operations should be used. When neglecting the fact that squaring in $\mathbf{Z}_n$ can be implemented somewhat faster than a general multiplication, elliptic curve addition and doubling operations require about 2 and 3 multiplications in $\mathbf{Z}_n$ and one division in $\mathbf{Z}_n$, respectively.

Division in $\mathbf{Z}_n$ can be implemented by the generalized Euclidean algorithm for computing greatest common divisors. The most efficient algorithm for computing multiplicative inverses, however, is that invented by Massey [17], which is a generalization of Stein's algorithm [25]. However, a division in $\mathbf{Z}_n$ seems to be less efficient than a multiplication in $\mathbf{Z}_n$.

On the other hand, if we calculate the addition on $E_n(a, b)$ in homogeneous coordinates, we can avoid the division in $\mathbf{Z}_n$ (except the final stage of the addition chain), although we must perform more multiplications instead.

Let $P_1 = (x_1, y_1, z_1) \in E_p(a, b)$, $P_2 = (x_2, y_2, z_2) \in E_p(a, b)$, and suppose that $P_1, P_2 \neq \mathcal{O}$, $P_1 \neq P_2$ and $P_1 \neq -P_2$. The addition formula [9] for $E_p(a, b)$ to find $P_3 = P_1 + P_2 = (x_3, y_3, z_3)$ is given by

$$\begin{cases} x_3 = v\{z_2(u^2 z_1 - 2v^2 x_1) - v^3\} \bmod p, \\ y_3 = z_2(3uv^2 x_1 - v^3 y_1 - u^3 z_1) + uv^3 \bmod p, \\ z_3 = v^3 z_1 z_2 \bmod p, \end{cases}$$

where $u = y_2 z_1 - y_1 z_2 \bmod p$, $v = x_2 z_1 - x_1 z_2 \bmod p$.

The doubling formula [9] for $E_p(a,b)$ to find $P_3 = 2 \cdot P_1$ is given by

$$\begin{cases} x_3 = 2y_1z_1(w^2 - 8x_1y_1^2z_1) \mod p, \\ y_3 = 4y_1^2z_1(3wx_1 - 2y_1^2z_1) - w^3 \mod p, \\ z_3 = 8y_1^3z_1^3 \mod p, \end{cases}$$

where $w = 3x_1^2 + az_1^2 \mod p$.

Then, one addition and doubling over $E_n(a,b)$ require 18 multiplications in $\mathbf{Z}_n$, if $a = 0$.

Therefore, in the affine coordinates, the computation amount for our scheme (Scheme 1) is about $(2+c)$ times as much as that for the RSA scheme, where c is the ratio of the computation amount of division in $\mathbf{Z}_n$ to that of multiplication in $\mathbf{Z}_n$. On the other hand, in the homogeneous coordinates, the computation amount for encryption with our scheme is about 18 times as much as that for the RSA scheme. Since in our elliptic curve system a message consists of two elements of $\mathbf{Z}_n$ compared to only one in the RSA system, the computation speed of our scheme is about $2/(2+c)$ or $1/9$ of the speed of RSA.

9 Conclusions

We have proposed new public key cryptosystems based on elliptic curves modulo n, where n is a product of two large primes. Furthermore, we have clarified the security of these systems. For the proposed Type 1 scheme, the master key concept [10] and the blind signature concept [2] are similarly applicable (using the combined techniques of Sections 7.4 and 7.5).

Acknowledgements

We would like to thank M. Benninger, Burt Kaliski, Neal Koblitz, Alfred Menezes, P. Schmid, and Hiroki Shizuya for their helpful discussions and support.

References

[1] Adleman, L., Manders, K. and Miller G.: "On taking roots in finite fields", *20th FOCS*, Vol. 20, 1977, pp.175-178 (1977).

[2] Chaum, D.: "Blind signatures for untraceable payments", *Proc. of Crypto'82* (1982).

[3] W. Diffie and M.E. Hellman, New directions in cryptography, *IEEE Transactions on Information Theory*, Vol. 22, No. 6, 1976, pp. 644-654.

[4] T. El-Gamal, A public key cryptosystem and a signature scheme based on the discrete logarithm, *IEEE Transactions on Information Theory*, Vol. 31, No. 4, 1985, pp. 469-472.

[5] S. Goldwasser and J. Kilian, Almost all primes can be quickly certified, *Proc. of the 18th Annual ACM Symposium on the Theory of Computing*, 1986, pp. 316-329.

[6] Hastad J.: "On using RSA with low exponent in a public key network", *Proc. of Crypto'85* (1985).

[7] Jonge W. and Chaum D.: "Attacks on some RSA signatures", Proc. of Crypto'85, (1985).

[8] B.S. Kaliski, A pseudo-random bit generator based on elliptic ligarithms, *Advances in Cryptology - CRYPTO '86*, Lecture Notes in Computer Science, Vol. 263, Berlin: Springer-Verlag, 1987, pp. 84-103.

[9] N. Koblitz, *A Course in Number Theory and Cryptography*, Berlin: Springer-Verlag, 1987.

[10] Koyama K.: "A Master Key for the RSA public-key cryptosystem", I.E.C.E. Trans.(D), J-65, 2, pp.163-170 (1982).

[11] E. Kranakis, *Primality and Cryptography*, Stuttgart: Teubner, and New York: John Wiley & Sons, 1986.

[12] A.K. Lenstra, H.W. Lenstra, M.S. Manasse and J.M. Pollard, The number field sieve, to appear.

[13] A.K. Lenstra and M.S. Manasse, Factoring with two large primes, *Advances in Cryptology - EUROCRYPT '90*, Lecture Notes in Computer Science, Berlin: Springer-Verlag, to appear.

[14] H.W. Lenstra, Factoring integers with elliptic curves, *Annals of Mathematics*, Vol. 126, 1987, pp. 649-673.

[15] U.M. Maurer, Fast generation of RSA-moduli with almost maximal diversity, *Advances in Cryptology - EUROCRYPT '89*, Lecture Notes in Computer Science, Vol. 434, Berlin: Springer-Verlag, 1990, pp. 636-647.

[16] U.M. Maurer and Y. Yacobi, Non-interactive public-key cryptography, to appear in Proc. Eurocrypt '91, Lecture Notes in Computer Science, Berlin: Springer-Verlag.

[17] J.L. Massey, Cryptography - fundamentals and applications (Copies of transparencies), Advanced Technology Seminars, Zurich, Switzerland, 1988.

[18] Miller, G. L. : "Riemann's hypothesis and tests for primality", J. Comput. System Sci. Vol.13, pp.300-317, (1976).

[19] A.J. Menezes, T. Okamoto, S.A. Vanstone, "Reducing Elliptic Curve Logarithms to Logarithms in a Finite Field", the Proceedings of STOC'91 (1991).

[20] M.O. Rabin, Probabilistic algorithm for testing primality, *Journal on Number Theory*, Vol. 12, l980, pp. 128-138.

[21] R.L. Rivest, A. Shamir, and L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, *Communications of the ACM*, Vol. 21, No. 2, 1978, pp. 120-126.

[22] R. Schoof, Elliptic curves over finite fields and the computation of square roots mod p, *Mathematics of Computation*, Vol. 44, No. 170, 1985, pp. 483-494.

[23] Silverman, J. H.: "The Arithmetic of Elliptic Curves ", Springer-Vaerlag, (1986).

[24] G. Simmons and M. Norris, Preliminary comments on the M.I.T public key cryptosystem, *Cryptologia*, Vol. 1, No. 4, Oct. 1977, pp. 406-414.

[25] J. Stein, Computational problems associated with Racah algebra, *J. Comp. Phys.*, Vol. 1, 1961, pp. 397-405.