ISOGENOUS ELLIPTIC CURVE CRYPTOSYSTEMS

ATSUKO MIYAJI

ABSTRACT. For a practical purpose, we often need a set of different elliptic curve cryptosystems with the same main performances and the same fundamental operations. In this paper, we show such a set of different elliptic curve cryptosystems exists. We also show such a set of different elliptic curve cryptosystems can be constructed in a practical time.

1. INTRODUCTION

From a security point of view, it is desirable that a cryptosystem, in each system or periodically in the same system, is changed to different cryptosystem which is secure if the original cryptosystem is attacked. Especially when we construct a cryptosystem with relatively low security level for the purpose of fast encryption/decryption, it is mandatory to change the cryptosystem periodically. From a practical point of view, it is desirable that we can change a cryptosystem by modifying only a few system parameters. Especially it is desirable that main performances of the cryptosystem (security level, running time and memory size) be fixed.

Elliptic curve cryptosystem, which was first proposed by Koblitz ([4]) and Miller ([6]), can offer a small key length cryptosystem if it avoids the Menezes-Okamoto-Vanstone reduction ([8]). The purpose of this paper is to study elliptic curve cryptosystem with the above desirable condition. We introduce the idea of *isogenous elliptic curves*, where elliptic curves over a finite field are called *isogenous* each other when they have the same number of rational points on the finite field ([10]). Furthermore we show that such an isogenous elliptic curve cryptosystem can be constructed in time

$$O((\log p)^{2+2k} L(\sqrt{p})^{2\sqrt{2}+O(1)}),$$

where $L(x) = \exp(\sqrt{\log x \log\log x})$. We also show that there exist many isogenous elliptic curve cryptosystems, each of which is constructed in the above time. Since an actual range of p for a secure elliptic curve cryptosystem over F_p is 100-bit or more, these results mean that we can offer enough many isogenous elliptic curve cryptosystems in a practical time.

This paper is organized as follows. Section 2 describes the ElGamal signature scheme on an elliptic curve and discusses the parameters which determines the main performances. Section 3.1 investigates the condition necessary to change cryptosystems. Section 3.2 describes elliptic curves which give different cryptosystems, keeping the main performances.

2. Elliptic curve cryptosystems

We will summarize cryptosystems using an elliptic curve over F_p, where $p \geq 5$. An elliptic curve over F_p is given as follows,

$$E : y^2 = x^3 + Ax + B \ \ (A, B \in F_p, 4A^3 + 27B^2 \neq 0).$$

Then the set of F_p-rational points on E (with a special element $\mathcal{O}$ at infinity), denoted $E(F_p)$, is a finite abelian group, where

$$E(F_p) = \{(x, y) \in F_p^2 | y^2 = x^3 + Ax + B\} \cup \{\mathcal{O}\}.$$

We show one example of elliptic curve cryptosystems, ElGamal Signature scheme and discuss the parameters of elliptic curve cryptosystems which determine main performances (security level, running time and memory size).

Let $m \in Z$ be a message. User A sends the message m to user B with her or his signature of m.

- Initialization
  - system parameter
    - $E : y^2 = x^3 + ax + b$ $(a, b \in F_p$; p is a prime of n bits).
    - $P \in E(F_p)$: a basepoint.
    - $l = ord(P)$ (l is t bits).
- Key generation
  User A randomly chooses an integer s as a secret key and makes public the point $P_A = sP$ as a public key.
- Signature generation
  1: User A picks a random number $k \in \{1, ..., l\}$ and computes

  $$R = kP = (r_x, r_y). \tag{1}$$

  Here $r_x = x(R)$ and $r_y = y(R)$.
  2: User A computes

  $$y \equiv \frac{m - sr_x}{k} \pmod{l} \tag{2}$$

  and outputs the signature (R, y).
- Signature verification
  1: User B checks that

  $$mP = yR + r_x P_A. \tag{3}$$

First as for the security level, it is determined by the size of the greatest prime divisor of l (the order of a basepoint) since the only known attacks for E/F_p chosen appropriately are the square root attacks. Therefore the parameter l determines the security level.

Next we discuss the fundamental operations which determine the running time. The most critical operation is an elliptic curve addition in (1) and (3). The addition is accomplished by the arithmetic on the definition field F_p (arithmetic modulo p). In the signature scheme, the division modulo l (the arithmetic modulo l) is also required. In a practical situation, the signature generation is often calculated on a smart card which has rather poor cpu power. Therefore we often compute (1) in off-line using idle-time. Thus the computation of (2) dominates the running time

of the signature generation. Except for the signature scheme, the fundamental operation of elliptic curve cryptosystems is determined by the arithmetic modulo p (definition field). Therefore the arithmetic modulo p and modulo l are regarded as fundamental operations of general elliptic curve cryptosystems such as ElGamal cryptosystem/signature, Diffie-Hellman etc.

Finally as for the memory size, it is clear that the data size (secret key, public key and system parameter) is determined by the size of l and p. To sum up, the parameters p and l determine the main performances.

3. Different Elliptic Curve Cryptosystems

We consider the following scenario: Each domain of an organization, which is independent and has equal relation with each other (for example, each laboratory of a company, each department of a company, each university in a country, etc.), needs a cryptosystem. In this scenario, the next conditions are often required:

1 Each domain requires a different cryptosystem with the same performances (security level, running time and memory size).
2 An user in a domain in an organization can communicate with or identify an user in the other domain in the organization easily.

From the above condition **1**, we want to construct a set of different elliptic curve cryptosystems in such way that, if one cryptosystem of the set is attacked, the rest are secure, and that they have the same performances. On the other hand, for the requirement **2**, any user in a domain must be able to use the other domain's cryptosystem easily. Therefore smaller differences between each cryptosystem are required. Especially, the fundamental operations of each cryptosystem must be same since any user in a domain cannot support many fundamental operations because of the limitation of the program size.

To sum up, we want a set of different elliptic curve cryptosystems with the same main performances and the same fundamental operations. Using such a set of different elliptic curve cryptosystems, we can change cryptosystems for each domain in an organization. This section demonstrates that elliptic curves over F_p can offer such cryptosystems easily.

We first discuss a necessary condition for a set of different elliptic curve cryptosystems and then investigate elliptic curves satisfying the condition.

3.1. Necessary condition for different elliptic cryptosystems. First we investigate the condition to offer a set of different elliptic curve cryptosystems. The only known attack on elliptic curve E/F_p chosen appropriately is to compute a logarithm for a given point. So the cryptosystem on E'/F_{p^9}, in which there does not exist an isomorphism to the original E/F_p calculated in a polynomial time, will be safe even if the cryptosystem on E/F_p should be attacked. An isomorphism between E'/F_{p^9} and E/F_p exists if and only if $p = p'$ and j-invariant of E, $j(E)$ equals $j(E')$. Therefore the condition of the different E'/F_{p^9} from the original E/F_p is either $p \neq p'$ or $j(E) \neq j(E')$ with $p = p'$.

Next we investigate the above condition in order to keep the same main performances and the same fundamental operations. As we have seen in Section 2, the parameters which influence on the main performances and fundamental operations

are p and l. So we want to keep these parameters fixed. As for the parameter l, it is fixed if the number of rational points is fixed. Combining the above discussion, we see that, for the original E/F_p, we need a different E'/F_p with $j(E) \neq j(E')$ and $\#E(F_p) = \#E'(F_p)$.

By Hasse's theorem, we have $|a| \leq 2\sqrt{p}$ for $a = p + 1 - \#E(F_p)$. Conversely, for any integer $|a| \leq 2\sqrt{p}$, there exists E/F_p with $\#E(F_p) = p + 1 - a$ ([2]). On the other hand, there are p elliptic curves over F_p modulo isomorphism. Therefore there exist a number of elliptic curves over F_p with a certain $\#E(F_p)$ points modulo isomorphism. Two elliptic curves E and E_1 over F_p are called isogenous if $\#E(F_p) = \#E_1(F_p)$ ([10]). From the above discussion, we need elliptic curves isogenous to the original elliptic curve modulo isomorphism.

3.2. Isogenous Elliptic Curve. In this section, we will describe the isogenous elliptic curves modulo isomorphism. For any $|a| \leq 2\sqrt{p}$, j-invariants of E/F_p with $p + 1 \pm a$ elements are represented as a solution of

$$\prod_{b^\theta | b} P_{Db^{\theta 2}}(X) \equiv 0 \pmod{p},\ 4p = a^2 + Db^2, \tag{4}$$

where $P_d(X)$ is a polynomial uniquely determined by d. For more information about this, we would refer the reader to [5]. An algorithm to construct the isogenous elliptic curves modulo isomorphism is obtained by generalizing the discussion [7].

Algorithm:

1: Choose a prime p.

2: Choose D with $\left(\frac{-D}{p}\right) = 1$.

3: Check $4p = a^2 + Db^2$ for an integer a, b. If such integers a and b do not exist, then goto step 2.

4: Set $N = p + 1 - a$ and $\tilde{N} = p + 1 + a$. Check either N or $\tilde{N}$ is divided by a large prime. If it is not divided, then goto step 2.

5: Calculate a class polynomial $P_{Db^{\theta 2}}(X)$ and solve $P_{Db^{\theta 2}}(X) \equiv 0 \pmod{p}$ for an integer b' with $b'|b$.

6: Take a solution j_0 of $P_{Db^{\theta 2}}(X) \equiv 0 \pmod{p}$. Construct an elliptic curves E/F_p with j-invariant j_0 and $\#E(F_p)$ equal to the one divisible by a large prime, N or $\tilde{N}$. Stop.

In step 5, the number of the solutions j of $P_{Db^{\theta 2}}(X) \equiv 0 \pmod{p}$ is equal to the degree of $P_{Db^{\theta 2}}(X)$, $deg(P_{Db^{\theta 2}}(X))$. Any solution j can give an elliptic curve with the required number of rational points, N or $\tilde{N}$. On the other hand, it is difficult to construct $P_{Db^{\theta 2}}(X)$ with a large Db'^2 since $deg(P_{Db^{\theta 2}}(X)) = O(\sqrt{Db'^2})$ (Siegel's result). Therefore we may choose a small D and set $b' = 1$ in order to construct only one elliptic curve over F_p as the original algorithm described in [7]. In this case, there is no problem on the running time of step 5.

Now we discuss the running time of Algorithm. In order to construct some different elliptic curve cryptosystems, we need compute step 5 for $b' > 1$. If $b' = O(\sqrt{p})$, then construction of $P_{Db^{\theta 2}}(X)$ requires $O(\sqrt{p})$ time. So we can not construct it in a practical time. Therefore the size of b' is important to the running time. So we add one more condition to step 3 of Algorithm: b is $L(\sqrt{p})^\alpha$-smooth. Here we call

an integer $L(x)^\alpha$-smooth when all of its prime factors are at most $L(x)^\alpha$, where

$$L(x) = \exp(\sqrt{\log x \log\log x}).$$

Then the probability that a random positive integer $b \leq \sqrt{p}$ is $L(\sqrt{p})^\alpha$-smooth is $L(\sqrt{p})^{1/(-2\alpha)+O(1)}$ for $p \to \infty$ ([1]). Here we assume that the probability holds for an integer b in step 5. In this case, Algorithm consists of three parts:

(1) Find D such that $4p = a^2 + Db^2$ for an integer a and b, that $N = p + 1 - a$ or $\bar{N} = p + 1 + a$ is divisible by a large prime, and that b is $L(\sqrt{p})^\alpha$-smooth (step 2, 3 and 4);
(2) For $b'|b$ with $b' \leq L(\sqrt{p})^\alpha$, construct a polynomial $P_{Db'^2}(X)$ and solve the equation $P_{Db'^2}(X) \equiv 0 \pmod{p}$ (step 5);
(3) Construct an elliptic curves E/F_p with j-invariant j_0 and the given number of rational points, where j_0 is a solution of $P_{Db'^2}(X) \equiv 0 \pmod{p}$ (step 6).

The expected running time of each step **(1)** $\sim$ **(3)** is analyzed as follows.
(1) The expected time needed to test a candidate is $O(\log^3 p)$ by using a probabilistic primality test ([9]). The expected number of repetition depends deeply on the product of the next two probabilities:
1. the probability that N or $\bar{N}$ is divisible by a large prime;
2. the probability that b is $L(\sqrt{p})^\alpha$-smooth.

From Cramer's conjecture, we assume that the former probability is $O(\log^{-k} p)$ for a positive integer k. Combining the probability of smooth integer, we see that the product is $O(L(\sqrt{p})^{1/(-2\alpha)+O(1)} \log^{-k} p)$. The remaining problem is the number of D necessary to be checked. It is reasonable to expect that we have to try roughly $O((\log^{2k} p)L(\sqrt{p})^{1/\alpha+O(1)})$ values of D by the following reason: For the bound B on D, we would expect that there are

$$\sum_{D=1}^{B} \frac{1}{2deg(P_D(x))} \geq \sum_{D=1}^{B} \frac{1}{2\sqrt{B}} = \frac{\sqrt{B}}{2}$$

values of D with $4p = a^2 + Db^2$. Therefore the expected final D is

$$O((\log^{2k} p)L(\sqrt{p})^{1/\alpha+O(1)})$$

and the expected time required in **(1)** is

$$O((\log^{2k+3} p)L(\sqrt{p})^{1/\alpha+O(1)}).$$

(2) Since the degree of $P_{Db'^2}(X)$ with the final D is

$$O(\sqrt{\log^{2k} pL(\sqrt{p})^{2\alpha+1/\alpha+O(1)}}),$$

the optimal choice of α is $\frac{1}{\sqrt{2}}$. So the expected time to construct a polynomial $P_{Db'^2}(X)$ is

$$O(\log^k pL(\sqrt{p})^{\sqrt{2}+O(1)}).$$

The remaining problem is to factorize $P_D(X)$ modulo p. It is computed in time $O((\log^{2+3k} p)L(\sqrt{p})^{3\sqrt{2}+O(1)})$ ([3]). Therefore the expected time required in **(2)** is

$$O((\log^{2+3k} p)L(\sqrt{p})^{3\sqrt{2}+O(1)}).$$

(3) The only problem in this stage is to determine which elliptic curve of at most 6 classes modulo F_p-isomorphism with a given j-invariant has the given number of rational points. Therefore the expected time required in **(3)** is $O(\log p)$.

Combining the above discussion, we conclude that the total expected time is

$$O((\log^{2+3k} p)L(\sqrt{p})^{3\sqrt{2}+O(1)})$$

for $p \to \infty$. In fact, we can construct isogenous elliptic curves in $deg(P_{Db^{\emptyset 2}}(x))$ numbers in this expected time. Therefore we can construct such an elliptic curve cryptosystem in time

$$O((\log p)^{2+2k} L(\sqrt{p})^{2\sqrt{2}+O(1)}).$$

Since we select D such that $4p = a^2 + Db^2$ and b is $L(\sqrt{p})^{1/\sqrt{2}}$-smooth, we can construct isogenous elliptic curves in the same time for the other $b"|b$ with $b" \neq b'$ and $b'' \leq L(\sqrt{p})^{1/\sqrt{2}}$. From the fact that the j-invariants of these elliptic curves are different each other ([2]), all of them satisfy the "necessary condition" in Section 3.1. Since an actual range of p for a secure elliptic curve cryptosystem over F_p is 100-bit or more, we conclude that we can construct enough many isogenous elliptic curve cryptosystems in a practical time.

In the case of F_{2^r}, we can construct E/F_{2^r} not by Algorithm but by the algorithm to find a suitable elliptic curve by computing the number of rational points of a randomly chosen elliptic curve. But in the algorithm, we can hardly construct an elliptic curve E_1 isogenous E since the probability to find such E_1 is too small. We often construct E/F_{2^r} by lifting an elliptic curve over a small field, like F_2 or F_4, to F_{2^r}. Even by this method, we can hardly construct such elliptic curves from the same reason.

4. Conclusions

From a security point of view, it is desirable that a cryptosystem, in each system or periodically in the same system, is changed to the different cryptosystem which is secure if the original cryptosystem is attacked. For this purpose, we have proposed isogenous elliptic curve cryptosystems which have the same parameters on the main performances and are secure if an elliptic curve cryptosystem is attacked. We have shown such an elliptic curve cryptosystem can be constructed in time

$$O((\log p)^{2+2k} L(\sqrt{p})^{2\sqrt{2}+O(1)}),$$

where $L(x) = \exp(\sqrt{\log x \log\log x})$. We also show that there exist enough elliptic curve cryptosystems, each of which is constructed in this time. Since an actual range of p for a secure elliptic curve cryptosystem over F_p is 100-bit or more, these results mean that we can offer enough many isogenous elliptic curve cryptosystems in a practical time.

References

1. E. R. Canfield, P. Erdös and C. Pomerance, On a problem of Oppenheim concerning "Factorisatio numerorum", *J. Number Theory*, 17(1983), 1-28.
2. M. Deuring, "Die Typen der Multiplikatorenringe elliptischer Funktionenkorper", *Abh. Math. Sem. Hamburg*, 14(1941), 197-272.
3. D. E. Knuth, *The art of computer programming, vol. 2, Seminumerical Algorithms*, 2nd ed., Addison-Wesley, Reading, Mass. 1981.
4. N. Koblitz, "Elliptic curve cryptosystems", *Mathematics of Computation*, 48(1987), 203-209.
5. S. Lang, *Elliptic Functions*, GTM112, Springer-Verlag, New York, 1987.
6. V. S. Miller, "Use of elliptic curves in cryptography", *Advances in Cryptology-Proceedings of Crypto'85*, Lecture Notes in Computer Science, 218(1986), Springer-Verlag, 417-426.
7. F. Morain, "Building cyclic elliptic curves modulo large primes", *Advances in Cryptology-Proceedings of Eurocrypt'91* , Lecture Notes in Computer Science, 547(1991), Springer-Verlag, 328-336.
8. A. Menezes, T. Okamoto and S. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field", *Proceedings of the 22nd Annual ACM Symposium on the Theory of Computing*, 80-89, 1991.
9. M. O. Rabin, "Probabilistic algorithms" in *Algorithms and complexity-new directions and recent results*, edited by J. F. Traub, Academic Press(1976).
10. J. H. Silverman, *"The Arithmetic of Elliptic Curves"*, GTM106, Springer-Verlag, New York, 1986.

Matsushita Electric Industrial Co., LTD., 1006, Kadoma, Kadoma-shi, 571 Japan
E-mail: miyaji@isl.mei.co.jp