Performances des cryptosystèmes basés sur les courbes elliptiques

Rakotondraina T. E.[1], Randimbindrainibe F.[2], Razakarivony J.[3]

Laboratoire d'Informatique appliquée, Images, Signal, Télécommunication, Automatique et Mathématiques appliquées (LIISTAM)

Département Télécommunication – Ecole Supérieure Polytechnique Antananarivo

Université d'Antananarivo
BP 1500, Ankatso – Antananarivo 101 - Madagascar

[1]*tahina.ezechiel@gmail.com*, [2]*falirandimby@yahoo.fr*, [3]*nainajd@yahoo.fr*

Résumé

Cet article compare deux cryptosystèmes à clé publique à savoir le cryptosystème RSA et le cryptosystème basé sur la courbe elliptique. Cette comparaison permet de mener une étude plus approfondie pour ce dernier. L'utilisation du logarithme discret comme algorithme de base sur ce cryptosystème basé sur la courbe elliptique permet d'obtenir un niveau de sécurité élevé pour une taille de clé moindre. En effet, la difficulté de la résolution du problème du logarithme discret, les caractéristiques et les propriétés des courbes elliptiques font que ce cryptosystème offre une optimisation au niveau de la taille des clés et des ressources allouées aux calculs et stockages des données. Ces résultats mènent vers une perspective sur une implémentation de ce cryptosystème sur les systèmes présentant des ressources limitées comme les cartes à puce et les réseaux sans fils.

Mots clés : Cryptographie, cryptosystème, courbe elliptique, logarithme discret, RSA

Abstract

This paper compares two cryptosystems using public-key: RSA cryptosystem and the cryptosystem based on the elliptic curve or Elliptic Curve Cryptosystem (ECC). This comparison allows doing a thorough study for the second case ECC. Using discrete logarithm as the basis of the algorithm, the cryptosystem based on the elliptic curve makes it possible to obtain high security level with a less size of key. Indeed, the difficulty of the resolution of the discrete logarithm problem, the characteristics and the properties of the elliptic curves make ECC sure and offer an optimization to the size-key level and resources allocated to calculations and data storages. These results lead to a prospect on an implementation of this cryptosystem on the restricted resource systems like the smart cards and the wireless networking.

Keywords: Cryptography, cryptosystem, elliptic curve, discrete logarithm, RSA, public-key

1. Introduction

La cryptographie est l'étude des techniques mathématiques relatives aux aspects de sécurité de l'information, telles celles concernant la confidentialité, l'intégrité et l'authentification des données ou de leur origine. Les algorithmes utilisés se séparent en deux groupes : les cryptosystèmes symétriques ou à clé secrète et les cryptosystèmes asymétriques ou à clé publique. C'est sur ce dernier que notre étude est fondée. La principale caractéristique du cryptosystème à clé asymétrique est l'utilisation d'une clé de chiffrement qui est publique et d'une clé de déchiffrement secrète connue seulement du récepteur.

Parmi les protocoles et cryptosystèmes qui furent développés au cours de l'histoire, nous allons en étudier deux types. Dans un premier temps, on développe le cryptosystème le plus utilisé RSA, pour Rivest, Shamir, Adleman [8], afin de connaître ses inconvénients ainsi que ses faiblesses. En deuxième lieu, notre étude est basée sur les courbes elliptiques, il permet une cryptographie efficace pour des tailles de clés moins longues que celles des autres protocoles asymétriques. La cryptographie basée sur les courbes elliptiques est née en 1985, découverte indépendamment par Miller V. [7] et Koblitz N. [4].

Notre recherche s'appuie sur les ouvrages Elliptic Curves Number Theory and Cryptography [10] et Guide to Elliptic Curve Cryptography [3].

Etudiées depuis des siècles, les courbes elliptiques sont parmi les objets mathématiques les plus complexes et les plus riches qui soient. Nous verrons une définition générale des courbes elliptiques ainsi que les opérations qui leurs sont associées, puis des algorithmes pour effectuer des calculs sur ces courbes le plus rapidement possible.

2. Moyens et Méthodes

2.1 Cryptosystème RSA

RSA est le cryptosystème le plus utilisé de nos jours. Son fonctionnement est le suivant :

Choisir deux nombres entier premier p et q, de l'ordre de 100 chiffres au minimum, pour rendre la factorisation hors de porter, même avec les meilleurs ordinateurs.

1) Calculer $n = p * q$
2) Calculer $m = (p - 1)(q - 1)$
3) Choisir un nombre entier e tel que $e > 2$ et $pgcd(e, n) = 1$
4) Calculer d tel que $d * e * mod\ m = 1$

On prendra comme clé publique e et n et comme clé privée d *et* n.

Connaissant la clé publique e et n, et le message à chiffrer M, on peut le chiffrer de la manière suivante :
Le message doit être remplacé par un chiffre. Ensuite on découpera le message par bloc de x longueurs avec $x < n$. Le bloc B est chiffré par la formule :

$$C = B^e * mod(n) \qquad (01)$$

C'est un bloc de message chiffré à envoyer vers le destinataire.

Pour le déchiffrement on va pratiquer quasiment de la même façon que le chiffrement mais avec la formule inverse :

$$b = C^d * mod(n) \quad (02)$$

Ce qui permettra au destinataire de trouver le message clair.

2.2 Définition d'une courbe elliptique

Soit K un corps, on appelle courbe elliptique sur K une courbe dans le plan projectif $\mathbb{P}^2(K)$, cubique et sans points singuliers, et munie d'un point distingué qui jouera un rôle particulier : élément neutre. Elle est donc définie par un polynôme irréductible homogène en trois variables à coefficient dans K [3]. Par un changement de variables homographique, on peut toujours se ramener à une équation dite de Weierstrass :

$$y^2z + a_1xyz + a_3yz^2 = x^3 + a_2x^2z + a_4xz^2 + a_6z^3$$

Avec $a_1, a_2, a_3, a_4, a_6 \in K$.

La courbe elliptique E est l'ensemble des points $(x, y) \in K^2$ satisfaisant cette équation et d'un point imaginaire $\mathcal{O}$ appelé point à l'infini [10].

2.3 Loi de Groupe

Les applications des courbes elliptiques en cryptographie sont principalement dues à l'existence d'une loi de groupe que nous pouvons définir sur ces dernières. En effet, l'ensemble $E \cup \mathcal{O}$ peut être équipé avec une opération d'addition qui produit un groupe abélien dont l'élément neutre est le point à l'infini $\mathcal{O}$ [10].

En cryptologie, les courbes elliptiques sont utilisées dans le corps $\mathbb{F}_p$ avec p un nombre premier strictement supérieur à 3.

Soit E une courbe elliptique définie sur $\mathbb{F}_p$. L'équation affine de Weierstrass de E peut être simplifiée, pour $K \neq 2,3$, par :

$$y^2 = x^3 + ax + b \quad (03)$$

La courbe définie par cette formule admet un unique point à l'infini (i.e. avec $z = 0$), de coordonnées (0 :1 :0). C'est en général ce point qui sera distingué. On appelle discriminant de cette courbe l'élément $-16(4a^3 + 27b^2)$ de K.

Le facteur entre parenthèse est le discriminant du polynôme membre de droite :

$$4a^3 + 27b^2 \not\equiv 0(mod\ p) \quad (04)$$

Soient $P = (x_1, y_1)$ et $Q = (x_2, y_2)$ deux points sur E :

- Le point $(x_1, -y_1)$ est l'opposé du point P et il est noté –P.
- Si $Q \neq P$ et $Q \neq -P$, alors le point $R = P + Q = (x_3, y_3)$ est défini par

$$\begin{cases} x_3 = \left(\frac{y_1 - y_2}{x_1 - x_2}\right)^2 - x_1 - x_2 \\ y_3 = y_1 + \left(\frac{y_1 - y_2}{x_1 - x_2}\right)(x_3 - x_1) \end{cases} \quad (05)$$

- Si $P = Q$, alors le point $2P = (x_3, y_3)$ est défini par :

$$\begin{cases} x_3 = \left(\frac{3x_1^2 + a}{2y_1}\right)^2 - 2x_1 \\ y_3 = x_1 + \left(\frac{3x_1^2 + a}{2y_1}\right)(x_3 - x_1) \end{cases} \quad (06)$$

- Si $x_1 = x_2$ mais $y_1 \neq y_2$, alors $R = \mathcal{O}$
- Si $P = Q$ et $y_1 = 0$, alors $R = \mathcal{O}$

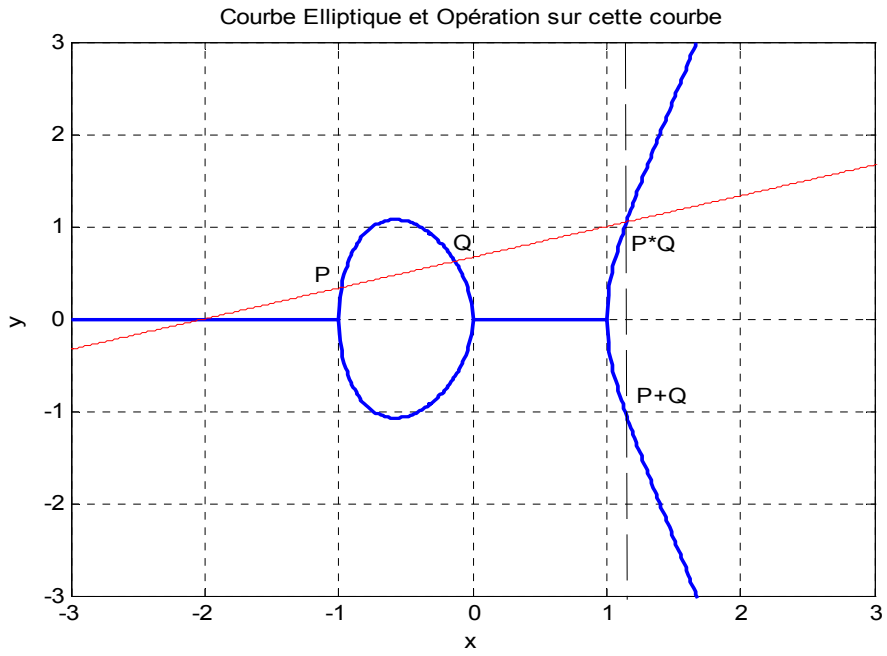

Figure 01 *: Courbe elliptique d'équation* $y^2 = 3x^3 - 3x$

Remarque : Notons $\mathcal{O}$ le point distingué de E. On montre qu'il existe une loi de groupe sur les points de E telle que $\mathcal{O}$ soit l'élément neutre et que l'identité $P + Q + R = \mathcal{O}$ soit vérifiée pour tout triplet de point alignés. Lorsque la convention $\mathcal{O} = (0:1:0)$ est en vigeur, l'opposé de chaque point de E est son symétrique par rapport à l'axe des abscisses [4].

La somme de deux points P, Q finis et non opposés est donc le point de coordonnées $(x_3, -y_3)$ *où* x_3, y_3 sont donnés par la relation (04). Bien sur, lorsque l'un des points est à l'infini, ou lorsqu' ils sont symétriques l'un de l'autre, on a les identités $P + \mathcal{O} = P$ et $P + (-P) = \mathcal{O}$.

2.4 Calcul du nombre de points d'une courbe elliptique sur un corps fini

Soit E une courbe elliptique d'équation (03) sur $\mathbb{F}_p$ (avec p premier).

Le nombre de points de E est donné par :

$$\#E = 1 + \sum_{x \in F_p} \left(1 + \left(\frac{x^3 + ax + b}{p}\right)\right) \qquad (07)$$

Si le corps de base est un corps fini, la courbe elliptique est un groupe fini et le théorème suivant donne un renseignement très utile sur son ordre [6].

Intéressons-nous sur les courbes elliptiques E sur un corps fini $\mathbb{F}_q$ (avec $q = p^r$).

$E(\mathbb{F}_q)$ a au maximum $2q + 1$ points c'est à dire le point à l'infini plus $2q$ paires $(x, y) \in \mathbb{F}_q^2$

Théorème 01 : (Théorème de Hasse)

Soit une courbe elliptique sur le corps fini $\mathbb{F}_q$. Le nombre de point de E vérifie :

$$q + 1 - 2\sqrt{q} \leq \#E \leq q + 1 + 2\sqrt{q} \qquad (08)$$

D'autre part, deux éléments suffisent pour engendrer le groupe d'une courbe elliptique et, bien souvent, il est même cyclique.

2.5 Problème du logarithme discret

Soit G un groupe (noté additivement) cyclique fini d'ordre N engendré par un élément P.

Soit Q un élément de G. Comme G est un groupe cyclique engendré par P, il existe un unique entier k compris entre 1 et N tel que $Q = kP$. Cet entier k est appelé le logarithme discret de Q en base P et nous le noterons $log_P(Q)$ [10].

Cependant le groupe que nous utiliserons par la suite est usuellement noté additivement et nous garderons donc les notations initiales. Notons bien que kP signifie $P + P + ... + P$ k fois où le signe « + » est la loi du groupe G.

La multiplication scalaire d'un point par un entier est l'opération de base et l'opération la plus chère des protocoles basés sur les courbes elliptiques.

Pour réduire le temps de calcul, on peut utiliser des algorithmes très connus d'exponentiation pour le calcul de $[k]P$ pour un k grand. On peut également améliorer le calcul pour un k petit.

2.6 Problème du logarithme discret sur les courbes elliptiques

Les cryptosystèmes utilisant les courbes elliptiques sont basés sur l'analogue du problème du logarithme discret sur les courbes elliptiques.

Soit E une courbe elliptique définie sur un corps fini $\mathbb{F}_q$ et soit P un point sur la courbe E.

Le problème du logarithme discret sur les courbes elliptiques (noté par ECDLP, Elliptic Curve Discrete Logarithm Problem) consiste à trouver un nombre k étant donné le point P et le point $Q = kP$ où $kP = P + P + \cdots + P$

Actuellement, le meilleur algorithme pour résoudre le ECDLP est en temps exponentiel en la taille de la clé k, en contraste avec les algorithmes sous-exponentiels connus pour factoriser les grands entiers. Ceci permet donc aux cryptosystèmes basés sur les courbes elliptiques d'utiliser, à sécurité équivalente, des clés beaucoup plus courtes que les cryptosystèmes asymétriques classiques comme RSA ou ElGamal [2].

2.7 Protocole d'échange de clés de Diffie-Hellman

Un des moyens pour sécuriser les données transitant entre l'émetteur et la récepteur est qu'ils établissent une clé privée entre eux. La méthode de Diffie-Hellman [1] permet justement de faire cela.

1) Alice et Bob choisissent une courbe elliptique E définie sur un corps fini $\mathbb{F}_q$ tel que le logarithme discret soit difficile à résoudre. Ils choisissent aussi un point $P \in \mathbb{F}_q$ tel que le sous-groupe généré par P ait un ordre de grande taille. (E et P sont choisis dont l'ordre soit un grand nombre premier.)

2) Alice choisit un nombre entier secret a, calcule $P_a = aP$ et envoie P_a à Bob.

3) Bob choisit un nombre entier secret b, calcule $P_b = bP$ et envoie P_b à Alice.

4) Alice calcule $aP_b = abP$ et Bob calcule $bP_a = baP$

5) Alice et Bob utilisent une méthode quelconque connue pour extraire une clé secrète de abP. Par exemple, ils peuvent utiliser les derniers 256 bits de la première coordonnée de abP comme clé, ou ils peuvent hacher une des coordonnées de abP avec une fonction de hachage pour laquelle ils se sont mis d'accord.

2.8 La taille des clés

Sur le groupe des points d'une courbe elliptique définie sur un corps fini, les meilleures attaques connues du logarithme discret sont des attaques génériques.

Lorsque l'on choisit une courbe, on essaie de trouver un groupe E tel que $\#E = h\mathcal{L}$ avec h petit et $\mathcal{L}$ premier.

Le point de base P qui servira à la multiplication $[k]P$ sera choisi dans le sous-groupe d'ordre $\mathcal{L}$.

Dans le cas où $h = 1, \#E = \mathcal{L}$. D'autre part, le théorème de Hasse dit que $\#E(\mathbb{F}_q) \sim q$.

Les attaques génériques (par exemple l'algorithme Pollard-Rho qui est une attaque par collision basée sur le paradoxe des anniversaires se font en $O(\sqrt{\mathcal{L}})$ [9].

Pour une sécurité de 2^{80} opérations, on voudra donc $\sqrt{\mathcal{L}} \geq 2^{80}$ soit $\sqrt{\mathcal{L}} \geq 2^{160}$, or $\#E(\mathbb{F}_q) \sim q$, pour $\#E(\mathbb{F}_q) \geq 2^{160}$, on choisira $q \geq 2^{160}$. Il nous faut donc travailler avec des corps d'ordre au moins égal à 2^{160}, c'est-à-dire travailler modulo un nombre premier d'au moins 160 bits dans le cas d'un corps premier.

Bien entendu, ce résultat est valable pour tous les niveaux de sécurité, c'est-à-dire que si l'on souhaite une sécurité de 2^n bits, il faut des clés de 2^{2n} bits.

RSA, qui n'est pas basé sur le problème du logarithme discret mais sur la factorisation en facteurs premiers, est sans doute le cryptosystème le plus utilisé mais pour une sécurité de 2^{80} opérations, il nécessite des clés de 1024 bits, c'est-à-dire des clés près de 7 fois plus longues que pour ECC.

Plus les niveaux de sécurité augmentent, plus ce rapport augmente.

Il est donc plus avantageux, en terme de taille de clés, d'utiliser ECC que RSA.

Cependant, si en termes de stockage, ECC est plus performant, on va voire après ce qu'il en est en termes de vitesse de calcul.

2.9 Inconvénients du RSA

Ce cryptosystème se réside sur la difficulté de factorisation d'un grand nombre entier. Le problème est que le niveau de sécurité s'accroit avec la taille de la clé qui est de l'ordre de 200 chiffres décimaux. La longueur de cette clé pose donc problème au niveau des calculs, de la vérification des clés, de l'authentification et du stockage des données.

3. Résultats et discussions

3.1 Calcul de la cardinalité

La cardinalité : Une courbe elliptique est considérée comme sûre lorsque sa cardinalité ou son ordre est quasi-premier.

Compter le nombre de points fait donc partie des problématiques importantes en cryptographie, c'est une étape indispensable dans la recherche de courbes cryptographiquement sûres.

Pour cela, on a mis au point un algorithme qui représente en entrée un nombre entier et avec une fonction *random()* le programme choisit au hasard un nombre premier inférieur à cet entier.

Les deux autres entrées sont les coefficients a et b et en faisant une boucle on vérifie le discriminant $4a^3 + 27b^2 \not\equiv 0 \ (mod\ n)$ d'après l'équation (04).

Ce premier programme permet donc d'obtenir un nombre premier, de vérifier le discriminant et en bouclant toutes ces données on a une cardinalité d'ordre quasi-premier.

Nous obtenons les résultats suivants pour une liste des nombres premiers inférieurs à 1000 compatibles avec, comme entrées, les coefficients a et b.

	-2 et 3					1 et 1					8 et 5			123 et 12			544 et 1454 (pour p < 1000)				
p	5	11	2	7	3	11	23	7	19	3	7	5	3	12	73	97	7	5	29	239	397
#E	5	5	2	5	2	7	13	5	11	3	2	2	2	11	61	97	7	3	23	227	389

Tableau 01 : *Quelques résultats de la cardinalité*

3.2 Calcul du temps de chiffrement

Dans tout système cryptographique il est nécessaire d'avoir un temps de chiffrement/déchiffrement suffisamment court. Quatre (4) algorithmes utilisant ECC ont été programmés et comparés par rapport à un programme de chiffrement avec RSA.

Ces quatre programmes se diffèrent par leur temps d'exécution qui est la conséquence de l'optimisation des algorithmes pour permettre d'éviter quelques attaques [5] mais aussi pour diminuer le temps de calcul.

Le chiffrement d'un mot présenté en entrée donne les proportions en temps (en seconde) suivantes pour chacun des algorithmes :

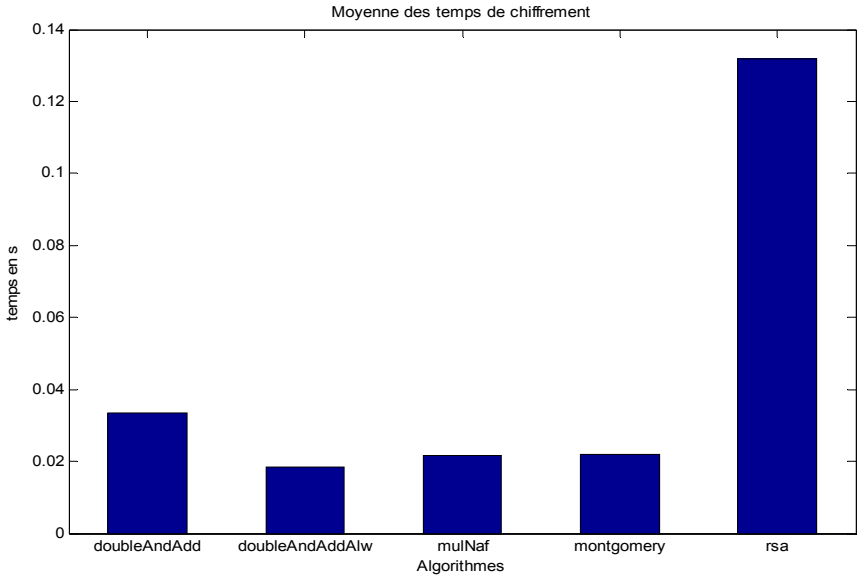

Figure 02 : *Estimation du temps de chiffrement des cinq algorithmes*

4. Conclusion et Perspectives

L'analyse de la performance apportée par la cryptographie basée sur la courbe elliptique nous mène vers une nouvelle perspective sur l'utilisation d'un tel cryptosystème. Par rapport au cryptosystème le plus utilisé RSA, la cryptographie basée sur les courbes elliptiques offre un niveau de sécurité élevée et surtout une performance du fait de la rapidité du temps de chiffrement des données. Vue la taille minime des clés pour un niveau de sécurité élevé, et l'optimisation du temps de calcul, il est donc intéressant d'envisager l'implémentation de l'ECC (Elliptic Curve Cryptosystem) sur les systèmes présentant des ressources en calcul et stockage de données limitées. On pourra ainsi optimiser et rendre plus performant un système utilisant une faible bande passante et nécessitant plusieurs opérations et calculs comme les cartes à puce, les réseaux sans fils.

5. Références

[1] W. Diffie, M. E. Hellman, « *New directions in cryptography* », IEEE Trans. Inform. Theory, vol. 22, n°6, pp. 644-654, 1976.

[2] T. ElGamal, *« A public key cryptosystem and a signature scheme based on discrete logarithms»*, IEEE Transactions on Information Theory, vol.31, pp.473-481, 1985.

[3] D. Hankerson, A. Menezes, Vanstone S, *« Guide to Elliptic Curve Cryptography »*, Springer, 2004.

[4] N. Koblitz, « *Elliptic curve crypto-systems»,* Math. of Comp., vol. 48 n°177, pp. 203-209, 1987.

[5] A.J. Menezes, P. C. Oorschot, S. A. Vanstone, *« Handbook of Applied Cryptography »*, CRC Press, 1996.

[6] A. J. Menezes, S. A. Vanstone, « *Elliptic curve cryptosystems and their implementation* ». Journal of Cryptology, vol. 6, pp. 209-224, 1993.

[7] V. S. Miller, « *Use of elliptic curves in cryptography*», Advances in Cryptology CRYPTO'85, Lecture Notes in Computer Science, pp. 417-426.

[8] R. L. Rivest, A. Shamir, L. Adleman, *« A method for obtaining digital signature and public key cryptosystems »*. Comm. ACM, vol. 21, pp.120-126, Feb 1978.

[9] J. Walter, *«The role of ECDSA in wireless communication (implement-tation and evaluation of ECDSA on constrained devices) »*, Los Angeles, 2002.

[10] L. C. Washington, *« Elliptic Curves Number Theory and Cryptography »*, Chapman & Hall/CRC, 2003.