Les Principes Théoriques De La Cryptographie Sur Les Courbes Elliptiques

Martin Rupp
SCIENTIFIC AND COMPUTER DEVELOPMENT SCD LTD

Un grand nombre d'algorithmes cryptographiques utilisent les courbes elliptiques (le plus connu d'entre eux est ECDSA (voir [6])). Ces algorithmes sont réputés plus sûrs que d'autres algorithmes équivalents à base de structures mathématiques qui sont "plus classiques". Ces algorithmes basés sur les courbes elliptiques peuvent consister en du chiffrement ou des signatures asymétriques mais les bases théoriques d'où elles proviennent sont souvent méconnues, voire totalement inconnues. En effet: quelles sont ces "courbes elliptiques", et en quoi assurent-elles une garantie de sécurité à ces algorithmes ? Et comment doivent-elles concrètement être implémentées.

Un rappel sur les courbes elliptiques

Generalités

Les courbes elliptiques sont des objets mathématiques très importants, notamment dans le domaine de la théorie des nombres ou de la géométrie algébrique. Les courbes elliptiques peuvent être très puissantes et elles ont été utilisées pour démontrer un certains nombres de théorèmes très difficiles comme particulièrement, plus récemment, le grand théorème de Fermat [ref].

Une courbe elliptique est - dans le cas complexe $\mathbb{C}$- une courbe isomorphe a une cubique de Weierstrass, généralement notée $\wp$, c'est à dire une courbe du plan projectif complexe d'équation:

$$Y^2 = 4X^3 - g_2X - g_3$$

et munie du point a l'infini.

Avec

$$(X, Y) \in \mathbb{C}$$

et

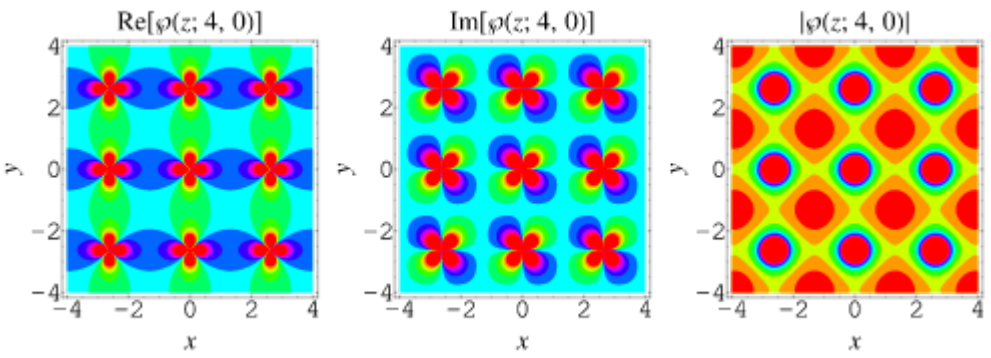
$$\Delta = -g_2^3 + 27g_3^2 \neq 0.$$

Une cubique de Weierstrass est associée à une fonction elliptique[1] $\wp(z, g_1, g_2)$qui la paramétrise rationnellement en utilisant à la fois $\wp(z)$ et $\wp'(z)$. Ces fonctions $\wp$ sont (doublement) périodiques. En effet elles sont invariantes sur un certain treillis Λ de $\mathbb{C}^2$.

La nature périodique peut être visualisée par des graphes de représentation avec $z = x + iy$:

$z \to Re(\wp(z))$
$z \to Im(\wp(z))$
$z \to |\wp(z)|$

[1] Elle aussi nommée fonction elliptique de Weierstrass



Les paramètres, g_2 et g_3 sont intimement liés au treillis Λ puisque l'on a :

$$\wp(z, g_1, g_2) = \frac{1}{z^2} + \sum_{\omega \in \Lambda} \left(\frac{1}{(z-\omega)^2} - \frac{1}{\omega^2} \right)$$

De manière assez intéressante, une courbe elliptique n'est donc nullement une ellipse (qui est une conique) mais bien en définitive *une cubique*.

Sur $\mathbb{R}$,une cubique de Weierstrass prend la forme caractéristique suivante:
Il s'agit d'une courbe symétrique par rapport à l'axe des abscisses

En fait, dans le cas général, une cubique $F(X, Y, Z)$ définie par les coordonnées projectives (X,Y,Z) sur un corps $\mathbb{K}$ **est une courbe elliptique** dès lors que:

1) F n'est pas singulière. C'est-à-dire que ses dérivées partielles n'ont aucun zéro en commun;

2) L'ensemble $\{(X,Y,Z)/F(X,Y,Z) = 0\}$ n'est pas vide.

Cette définition est la plus générale et permet de définir les courbes elliptiques dans les corps finis qui sont évidemment ceux qui peuvent être utilisés dans le cadre d'applications cryptographiques.
Notons que $F(X,Y,Z)$ peut définir une courbe elliptique sur un corps $\mathbb{K}$ mais pas sur un autre corps $\mathbb{K}'$.

Par exemple $F(X,Y,Z) = X^3 + 7Y^3 + 49Z^3$ n'est pas une courbe elliptique sur $\mathbb{Q}$ mais par contre c'est une courbe elliptique sur $\mathbb{Q}[\sqrt[3]{7}]$. En effet, l'ensemble $\{F(X,Y,Z) = 0\}$ est vide sur $\mathbb{Q}$ mais pas sur $\mathbb{Q}[\sqrt[3]{7}]$: sur ce corps, $M = (0, -\sqrt[3]{7}, 1)$ ou $N = (-7/\sqrt[3]{7}, 0, 1)$ sont, entre autres, des points de la courbe.

Une correspondance rationnelle ϕ peut être définie sur des courbes elliptiques (comme sur des courbes algébriques en général) par la formule:

$$\phi(x,y,z) = (A(x,y,z), B(x,y,z), C(x,y,z))$$

Avec les conditions suivantes:

1) A,B,C sont des polynômes rationnels de degré d.
2) Pour presque tous les points[2] :
$(x,y,z) \in F(\overline{\mathbb{K}}) \implies \phi(x,y,z) \in G(\overline{\mathbb{K}})$

Une équivalence birationnelle entre deux courbes F, G existe lorsque deux deux équivalences rationnelles existent ϕ: $F \to G$ et φ: $G \to F$ et qu'elles sont les inverses l'une de l'autre. Dans ce cas, elles sont dites *équivalentes*.

Lois de groupe sur une courbe elliptique

Cas complexe

Sur $\mathbb{C}$, il y a une équivalence entre $\mathbb{C}/\Lambda$ et la courbe elliptique définie par la cubique de Weierstrass associée au treillis Λ. Cette équivalence est définie par une bijection f qui est en fait la *paramétrisation* de la courbe elliptique mentionnée précédemment.

- $z \to f(z) = (\wp(z), \wp'(z), 1)$ sur $z \in \mathbb{C}/\Lambda, z \notin \Lambda$
- $f(\Lambda) = (0,1,0)$

Cette paramétrisation transporte la structure de groupe de $\mathbb{C}/\Lambda$ sur la courbe elliptique associée.

[2] e.g. a part, peut-être, pour un nombre fini d'entre eux

Pour pouvoir calculer l'addition de deux points A et B de la courbe elliptique , par exemple: $C = A \oplus B$, il faut utiliser la structure de groupe sur $\mathbb{C}/\Lambda$:

$$A = f(u),\ B = f(v).\ A \oplus B = f(u + v).$$

C'est-a-dire:

$$A \oplus B = (\wp(u+v), \wp'(u+v), 1)$$

Il y a une visualization géométrique, dans le plan projectif $\mathbb{P}_2(\mathbb{C})$, de la construction de $A \oplus B$: on prouve que A, B, C sont colinéaires si et seulement si $A \oplus B \oplus C = 0$ ([1] §2.8) , comme selon le théorème de Bezout il ne peut pas y avoir plus de trois points colinéaires sur une courbe elliptique (puisqu'elle est de degré 3) , on peut (schématiquement) représenter une construction de la loi de groupe sur une courbe elliptique.

Ici $C = -(A \oplus B)$, $A \oplus B$ est donc obtenu comme étant le symétrique de C par rapport à l'axe de symétrie de la courbe elliptique (les courbes de Weierstrasses sont toujours symétriques) . L'élément 0 est le point a l'infini.

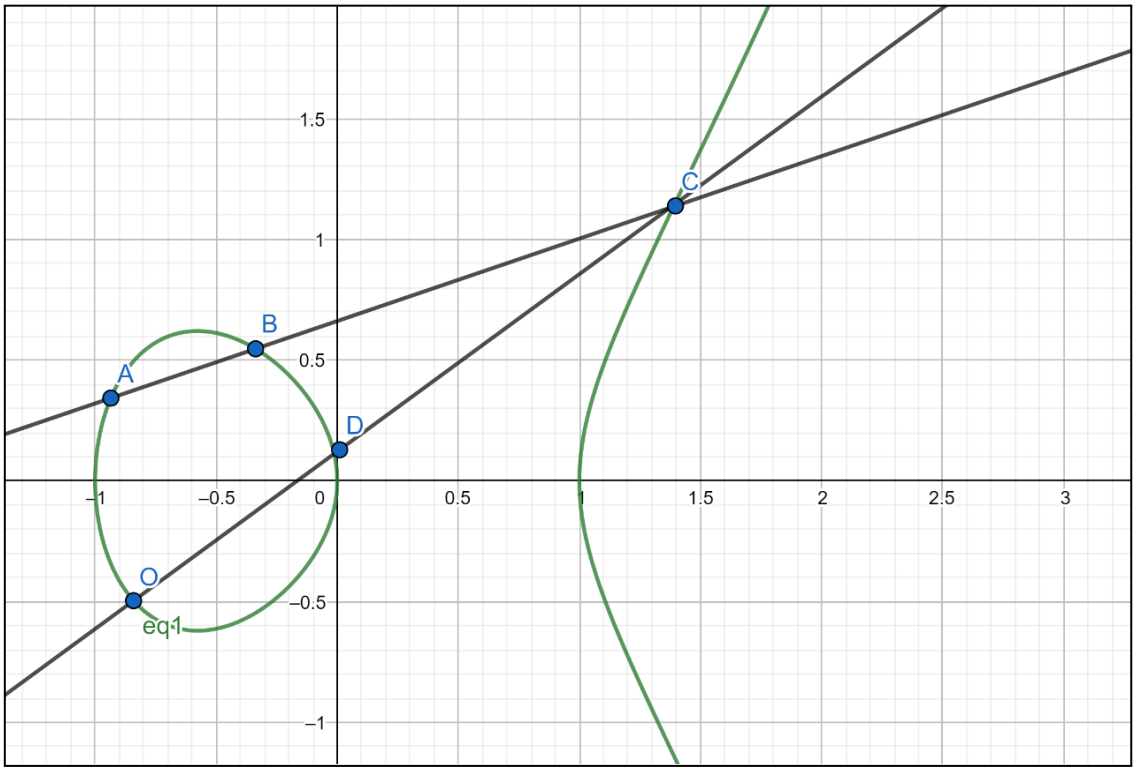
Une représentation graphique directe est possible sur $\mathbb{R}$ mais pas sur $\mathbb{C}$. Néanmoins la construction de $A \oplus B$ s'obtient sur $\mathbb{C}$ par les étapes suivantes:

1) Si A et B ne sont pas symétriques, construire le point C comme l'intersection de la courbe elliptique avec la droite (AB)
2) Construire le symétrique de C sur la courbe elliptique

Cas général

Dans le cas général, on suppose la courbe sans points multiples et l'élément 0 est choisi comme un point arbitraire O de la courbe elliptique $C(\mathbb{K})$.

Soit A et B deux points de $C(\mathbb{K})$. La ligne (AB) coupe la courbe elliptique en un troisième point C. De même, la ligne (OC) coupe la courbe en un point D. On définit D comme $A \oplus B$.



On vérifie que, pour tout point A (y compris O) on a bien:
$O \oplus A = A$
Et on vérifie les propriétés de la structure de groupe.
$(C, O, \oplus)$ est un groupe Abélien et on a un isomorphisme de groupe
$(C, O, \oplus) \sim (C, O', \oplus)$ pour deux points-origine quelconques O et O'.

Deux notions sont importantes pour travailler avec les courbes elliptiques:

1) Les isogénies de courbes elliptiques: Une isogénie φ définie entre deux courbes elliptiques (C_1, O_1) et (C_2, O_2) est une application rationnelle de C_1 vers C_2 telle que $\varphi(O_1) = O_2$.
2) La hauteur d'une courbe elliptique. C'est un cas particulier d'une fonction de type hauteur définie pour un groupe abélien. Il existe plusieurs hauteurs sur les courbes

elliptiques dont la hauteur Logarithmique, la hauteur canonique et la hauteur de Néron-Tate.

Formule d'addition:

Si $P = (x_P, y_P)$ et $Q = (x_Q, y_Q)$ et $y^2 = x^3 + ax + b$
Nous calculons R=P+Q:
$y' = s(x' - x^{q^2}) - y^{q^2}$
Si $x_p \neq x_Q$:

- $s = \frac{y_P - y_Q}{x_P - x_Q}$ *(pente)*

Si $x_P = x_Q$:

- $s = \frac{3x_P^2 + a}{2y_P}$ *(tangente)*
- $a = \frac{y_Q^2 - y_P^2 - (x_Q^3 - x_P^3)}{x_Q - x_P}$

Dans les deux cas:

- $y_R = s(x_R - x_P) - y_P$
- $x_R = s^2 - (x_P + x_Q)$ ($x_R = s^2 - 2x_P$ dans le cas $x_P = x_Q$)

Les courbes Elliptiques sur les corps finis

Dans le cadre des applications cryptographiques, on s'intéresse aux courbes elliptiques sur les corps finis. Soit donc $\mathbb{F}_q$ un corps fini et $E = E(\mathbb{F}_q)$ une courbe elliptique sur ce corps définie par une équation de Weierstrass:

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$$

Il est important de connaître l'ordre de E, c'est-a-dire de calculer $\#E(\mathbb{F}_q)$.
Un théorème de Hasse permet d'approximer cet ordre par q+1:

$$|\#E(\mathbb{F}_q) - (q+1)| \leq 2\sqrt{q}$$

L'algorithme de Schoof permet de compter exactement , en un temps polynomial, les points d'une courbe elliptique.

L’algorithme de Schoof

L’algorithme de Schoof se base sur le fait que l’on a une isomorphie bien connue de $\mathbb{Z}/N\mathbb{Z}$ avec un produit de groupes cycliques:

$$\mathbb{Z}/N\mathbb{Z} \sim \prod_{i=1\ldots n} \mathbb{Z}/n_i\mathbb{Z}$$

pour $N = n_1 \ldots n_n$ avec les n_i sans facteurs communs [3]

En utilisant le théorème de Hasse, il suffit donc de calculer $t \pmod N$ avec

$$t = \#E(\mathbb{F}_q - (q+1) \text{ et } N > 4\sqrt{q}$$

Pour calculer cette quantité, on peut calculer individuellement des quantités $t \pmod{l_i}$ avec:

$$N = \prod_i l_i \quad .$$

et les l_i sont des entiers premiers.

Ensuite, l’algorithme utilise l’endomorphisme de Frobenius $\phi : (x, y) \to (x^q, y^q)$. Cet endomorphisme satisfait a l'équation quadratique (I est l'application identité):

$$\phi^2 - t\phi + qI = 0$$

Il suffit donc de résoudre l'équation:

$$\phi^2(M) - t\phi(M) + qM = 0$$

Pour un point M quelconque de la courbe

.

La technique de Schoof consiste donc à calculer:

$$t_i = t \pmod{l_i}$$

Pour certains entier premiers l_i.

On constate que si l’on choisit un point de la courbe $P = (x, y)$ dans le sous-groupe de Torsion $E[l_i] = \{P \in E, l_i P = O\}$, alors $qP = q_i P$, $q_i = q \pmod{l_i}$ et $|q_i| < l_i/2$

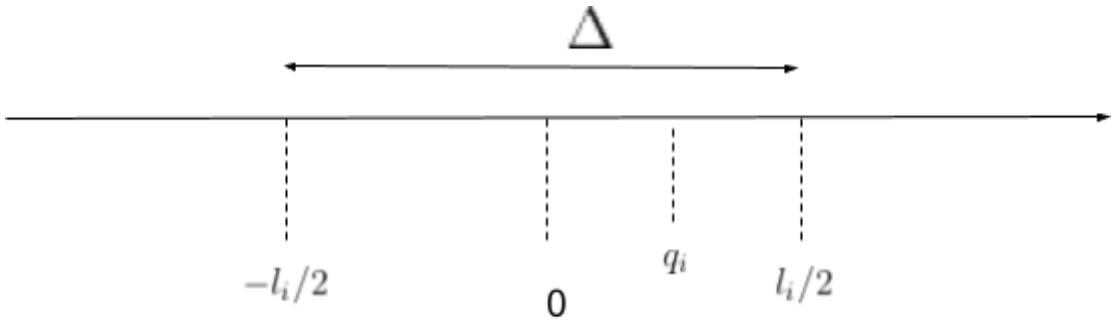
.

On peut toujours trouver un tel nombre car sur n’importe quel interval Δ de longueur $\geq l_i$ il existe un entier q_i tel que:

---

[3] Théorème des restes chinois

- $q_i = q \pmod{l_i}$
- $q_i \in \Delta$ (cf ci-dessous)



Puisque $\phi(l_i P) = l_i \phi(P) = 0$ si $P \in E[l_i]$, on a $t\phi(P) = t_i \phi(P)$ avec $t_i = t \pmod{l_i}, |t_i| < l_i/2$. On doit donc résoudre l'équation:

$$\phi^2(M) - t_i \phi(M) + q_i M = 0$$

Si l'on definit $\chi(M)$ par:

$$\chi(M) = \phi^2(M) + q_i M$$

Le calcul de t_i se fait en résolvant l'équation:

$$\chi(M) = t_i \phi(M), t_i \in \mathbb{Z}$$

Il y a deux cas a distinguer:

1) $\phi^2(M) \neq \pm q_i M$ et
2) $\phi^2(M) = \pm q_i M$

On doit distinguer ces deux cas car l'addition de deux points A,B sur une courbe elliptique se fait obligatoirement de trois manières différentes suivant que:

- $A = B$
- $A = -B$
- $A \neq \pm B$

<u>Cas 1)</u>

Il est possible de calculer explicitement $X(M)$ en utilisant la formule d'addition sur les courbes elliptiques (décrite précédemment) :

On note $M = (x, y)$, $q_i M = (x_i, y_i)$, $\chi(M) = (x', y')$, $t_i \phi(M) = (x_{t_i}, y_{t_i})$

x_i, y_i peuvent êtres calculés par les polynômes de divisions ϕ_q, φ_q, ω_q (voir ci-après)
D'autre part, on a :

$$\phi^2(M) = (x^{q^2}, y^{q^2})$$

En utilisant la formule d'addition, on a :

- $s = \dfrac{y^{q^2} - y_i}{x^{q^2} - x_i}$
- $x' = (\dfrac{y^{q^2} - y_i}{x^{q^2} - x_i})^2 - (x^{q^2} + x_i)$
- $y' = s(x' - x^{q^2}) - y^{q^2}$

Si nous essayons de résoudre $x' = x_{t_i}$, il y aura uniquement deux choix possibles pour y' , identiques a un signe près.

x' est une fonction de x uniquement puisque $(y^{q^2} - y_i)^2$ peut se factoriser en

$$y^2((y^2)^{\frac{q^2-1}{2}} - (y_i/y)^2)^2 = (x^3 + Ax + B)[(x^3 + Ax + B)^{\frac{q^2-1}{2}} - \theta(x)]^2$$

Quant a x_i, il se calcule avec les polynômes de division ψ_n:

$$x_i = x - \frac{\psi_{q-1}\psi_{q+1}}{\psi_q^2}(x)$$

C'est aussi une fonction de x uniquement.

A ce stade, nous devons résoudre une équation à une inconnue en t_i :

$$x' = x_{t_i}$$

t_i peut prendre les valeurs $1, 2, \cdots, \frac{l_i - 1}{2}$.

Cette équation est équivalente à :

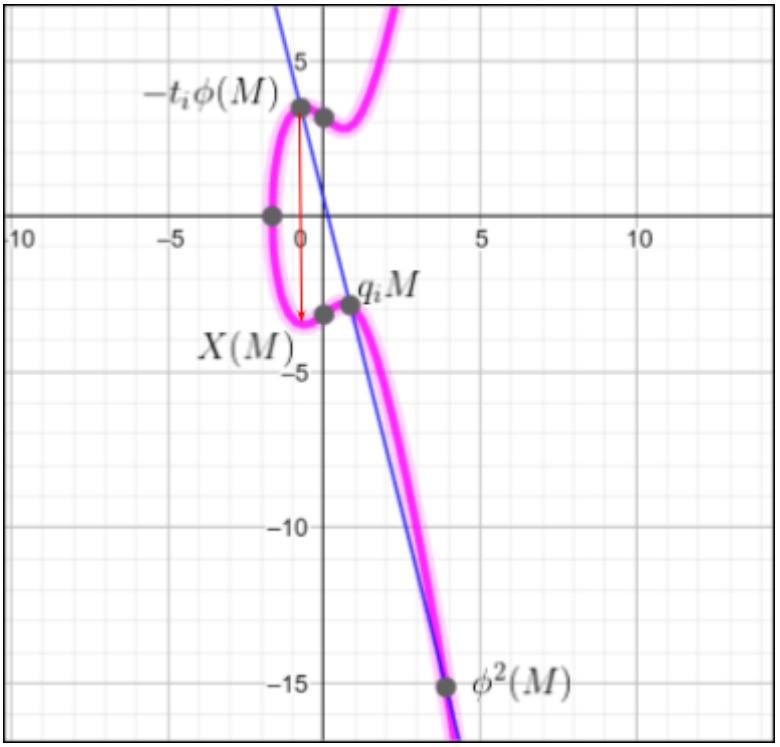
$$x' = x_{t_i} \pmod{\psi_{l_i}}$$

x_{t_i} se calcule en utilisant les polynômes de divisions et ,généralement, en calculant les valeurs pour des valeurs de $q = 2^k$ et le fait que modulo ψ_{l_i} on a :

$x_{t_i}(q) = x_{t_i}(2^{k_1}) \times \ldots \times x_{t_i}(2^{k_r})$ avec $q = 2^{k_1} + \ldots + 2^{k_r}$, la décomposition binaire de q.

L'algorithme de Schoof consiste donc à ce stade à calculer les candidats possible de t_i et à s'arrêter dès qu'une valeur est trouvée. Il reste ensuite à tester deux valeurs maximum pour y'.

Bien sur on ne sait pas qu'on est dans le cas '1' au départ donc si l'algorithme échoue, on se retrouve dans le second cas.



Cas 2)

Le cas 2) est différent du cas 1) car $\phi^2(M)$ et q_iM sont désormais symétriques l'un de l'autre ou confondus, ce qui fait, en réalité, deux sous-cas.

L'algorithme procède d'une manière similaire au cas 1.
On montre rapidement que, dans ce cas ou les points sont confondus:

$$t_i^2 q_i^2 = (2q)^2 \pmod{l_i}$$

Ce qui se résout relativement facilement en calculant les racines carrées de q sur $\mathbb{F}_l$.

Dans le cas où de telles racines carrées n'existent pas, c'est que les points sont symétriques et la aussi le calcul est effectué rapidement.

L'algorithme de Schoof n'est concrètement pas très pratique et efficace et il a été amélioré par l'algorithme SEA (Schoof–Elkies–Atkin algorithm) qui est généralement celui qui est utilisé dans les implémentations.

Le calcul du nombre de points d'une courbe elliptique est primordial pour pouvoir programmer les algorithmes cryptographiques utilisant les courbes elliptiques.

Cryptographie sur les courbes elliptiques

Le problème du logarithme discret sur les courbes elliptiques

La résolution du logarithme discret est un problème difficile (cf §2.6,[3]) . Puisqu'il est possible de définir une structure de groupe sur une courbe elliptique $E = E(\mathbb{F}_q)$, la résolution du logarithme discret y est également possible:

Supposons connu $(g, Y) \in E^2$, peut-on trouver un entier x tel que :
$g^x = Y$ (ou $xg = Y$ si on note la loi de groupe additivement)

Dans les courbes elliptiques, ce problème est particulièrement dur. Les algorithmes tels que le calcul d'index permettant de résoudre le problème du logarithme discret "classique" ne peuvent pas être appliqués dans le cadre des courbes elliptiques. Malgré des progrès récents (cf [4]) sur les corps binaires (e.g $\mathbb{F}_{2^k}$) , le logarithme discret ne peut être pour le moment calculé qu'avec des temps de calculs exponentiels.
Dans certains types de courbes elliptiques, commes les courbes 'anormales'
(e.g les courbes telles que : $\#E(\mathbb{F}_q) = q$) , le problème du logarithme discret peut être résolu ([7]). Il est donc important, lors de l'implémentation des algorithmes, de ne choisir que des courbes elliptiques 'sûres'.

Le problème du logarithme discret dans les courbes elliptiques peut également se formuler à l'aide des polynômes de divisions ψ_n (voir [8], §3.2 pour leur définition) . Connaissant les points P et Q , peut-on trouver n tel que $Q = nP$?

Ce qui se traduit par le système (ou n est la seule inconnue):

- $P = (x, y)$
- $Q = (u, v)$
- $u = \dfrac{\phi_n(x)}{\psi_n^2(x)}$

- $v = \dfrac{\omega_n(x,y)}{\psi_n^3(x,y)}$
- $y^2 = x^3 + Ax + B$
- $u^2 = v^3 + Av + B$
- $\phi_n(x,y) = \phi_n(x) = x\psi_n^2(x,y) - \psi_{n-1}(x,y)\psi_{n+1}(x,y)$
- $\omega_n(x,y) = \dfrac{1}{4y}(\psi_{n+2}(x,y)\psi_{n-1}^2(x,y) - \psi_{n-2}(x,y)\psi_{n+1}^2(x,y))$

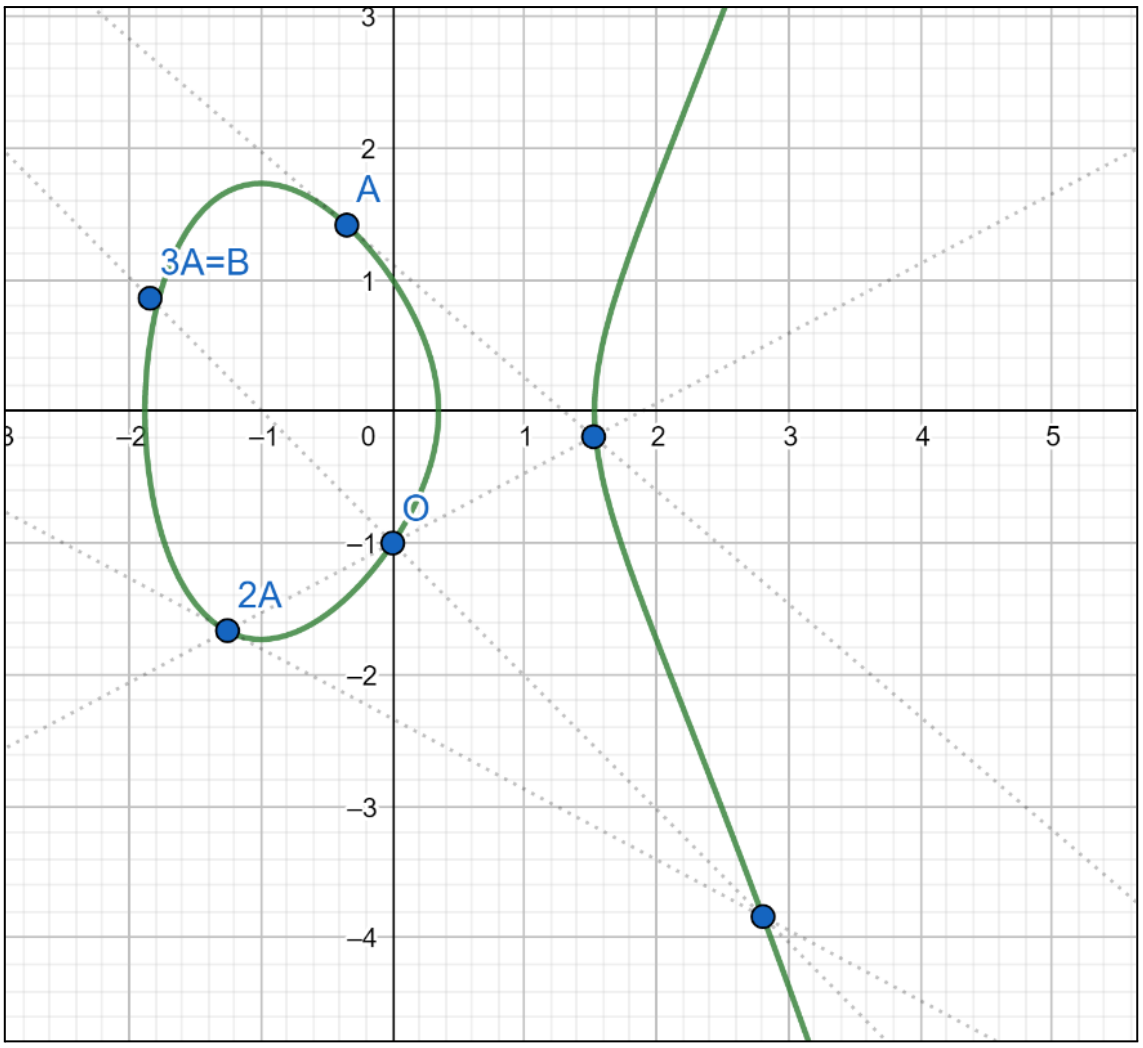
Notons que $\psi_n^2(x,y) = \psi^2(x)$ car cette expression ne dépend pas concrètement du terme en y (et cela est vrai pour d'autres expressions similaires pour les autres fonctions considérées ci-dessus)

Ce système ne permet pas néanmoins de résoudre le logarithme discret directement mais il permet le calcul concret dans le cas d'approche brute force ou 'pas de bébé'-'pas de géant'.

Parmi les algorithmes permettant de calculer le logarithme discret dans les corps finis (cf [5]) , citons:

- Methodes de calcul d'index;
- Baby Step-Giant Step ('pas de bébé'-'pas de géant');
- Pohlig–Hellman;
- Algorithme rho de Pollard pour les logarithmes.

L'exemple ci-dessous illustre la résolution du problème du logarithme traité de manière géométrique. Par exemple ici 3A=B, le problème est résolu pour A et B.



Algorithmes Cryptographiques

La cryptographie sur les courbes elliptique est en générale utilisée via trois algorithmes:

- Le protocole d'échange de clef Diffie-Hellman (ECDH);
- La signature ECDSA , une variante de l'algorithme de signature DSA utilisant les courbes elliptiques;
- Le système de chiffrement El-Gamal a base de courbes elliptiques (EC-ELGAMAL).

Il y a néanmoins beaucoup d'autres algorithmes cryptographiques utilisant les fonctions elliptiques.

ECDH

L'algorithme d'échange de clefs Diffie-Hellman est relativement simple et peut être décrit comme suit sur $\mathbb{Z}/n\mathbb{Z}$:

DIFFIE-HELLMAN

Alice	Environnement Public	Bob
Accord sur les paramètres communs g, n		Accord sur les paramètres communs g, n
Choix d'un entier aléatoire y		Choix d'un entier aléatoire x
Calcul de $Y = g^y(n)$		Calcul de $X = g^x(n)$
	$\leftarrow X$	
	$Y \rightarrow$	
Calcul de $K_2 = X^y(n)$		Calcul de $K_1 = Y^x(n)$
$K_1 = K_2 = K$		$K_1 = K_2 = K$

Ici, la clef commune est *K*.

Sur une courbe elliptique (L'algorithme se nomme alors ECDH), il peut être décrit comme suit:

ECDH

Alice	Environnement Public	Bob
Accord sur les paramètres communs $\mathbb{F}_q, E(\mathbb{F}_q), P \in E(\mathbb{F}_q)$		Accord sur les paramètres communs $\mathbb{F}_q, E(\mathbb{F}_q), P \in E(\mathbb{F}_q)$
Choix d'un entier aléatoire y		Choix d'un entier aléatoire x
Calcul de $Y = yP \in E(\mathbb{F}_q)$		Calcul de $X = xP \in E(\mathbb{F}_q)$
	$\leftarrow X$	
	$Y \rightarrow$	
Calcul de $K_2 = yX \in E(\mathbb{F}_q)$		Calcul de $K_1 = xY \in E(\mathbb{F}_q)$
$K_1 = K_2 = K$		$K_1 = K_2 = K$

Dans le contexte des courbes elliptiques, les entiers x,y sont les clefs privées alors que les points (X,Y) sont les clefs publiques. La clef commune *K* , échangée, est un point de la courbe elliptique.

Dans les deux cas, DH et ECDH, la sécurité repose (presque) entièrement sur la difficulté de résoudre le problème du logarithme discret.

ECDSA

DSA

Alice	**Environnement Public**	**Bob**
Accord sur les paramètres communs H, N, L, q, p H: function de hash L: key length N: modulus $(N < L,\ N < \|H\|)$ q: N-bits p: L-bits $p - 1 = 0(q)$		Accord sur les paramètres communs H, N, L, q, p H: function de hash L: key length N: modulus $(N < L,\ N < \|H\|)$ q: N-bits p: L-bits $p - 1 = 0(q)$
	← Génération aléatoire d'un entier h →	
Calcul de $g = h^{(p-1)/q}(p)$ $g \neq 1$		Calcul de $g = h^{(p-1)/q}(p)$ $g \neq 1$
	Paramètres de l'algorithme: (p,q,g)	
Generation de Clefs		
Calcul de la paire de clef $\{y_A, x_A\}$ Choix aleatoire de $x_A \in [1, q-1]$ $y_A = g^{x_A}(p)$		Calcul de la paire de clef $\{y_B, x_B\}$ Choix aleatoire de $x_B \in [1, q-1]$ $y_B = g^{x_B}(p)$
x_A : clef privee		x_B : clef privee
	Publication de y_A comme clef publique associée à Alice	
	Publication de y_B comme clef publique associée à	

	Bob	
Signature d'un message m		
Choix de $k \in [1, q-1]$		
$r = (g^k(p))(q)\ r \neq 0$		
$s = (k^{-1}(H(m) + x_A r))(q)$ $s \neq 0$		
	$(m, r, s) \rightarrow$	
		Lecture du message m de la signature (r,s)
Vérification d'un message m		
		Verifier que $(0 < r < q)$
		Verifier que $(0 < s < q)$
		Calcul de $w = s^{-1}(q)$
		Calcul de $u_1 = H(m).w\,(q)$
		Calcul de $u_2 = r.w\,(q)$
		$v = ((g^{u_1} y_A^{u_2})(p))(q)$
		Verifier que $v = r$

ECDSA

Alice	**Environnement Public**	**Bob**
Accord sur: $\mathbb{F}_p$, une courbe elliptique $E(\mathbb{F}_p)$, un point $G \in E(\mathbb{F}_p)$ (d'ordre 'tres important') H: function de hash		Accord sur: $\mathbb{F}_p$, une courbe elliptique $E(\mathbb{F}_p)$, un point $G \in E(\mathbb{F}_p)$ (d'ordre 'tres important') H: function de hash
Generation de Clefs		
Calcul de la paire de clef $\{y_A, x_A\}$ Choix aleatoire de		Calcul de la paire de clef $\{y_B, x_B\}$ Choix aleatoire de

$x_A \in [1, q-1]$ $Y_A = x_A G$		$x_B \in [1, q-1]$ $Y_B = x_B G$
x_A : clef privee		x_B : clef privee
	Publication de Y_A comme clef publique associée à Alice	
	Publication de Y_B comme clef publique associée à Bob	
Signature d'un message m		
Choix de $k \in [1, q-1]$		
Calcul de $kG \in E(\mathbb{F}_p)$		
Calcul de $r = x(kG)(q)$ ($P \to x(P)$ est la fonction de projection sur l'axe des abscisses) . Vérifier que r n'est pas le point a l'infini.		
$s = (k^{-1}(H(m) + x_A r))(q)$ $s \neq 0$		
	$(m, r, s) \to$	
		Lecture du message m de la signature (r,s)
Vérification d'un message m		
		Verifier que $(0 < r < q)$
		Verifier que $(0 < s < q)$
		Calcul de $w = s^{-1}(q)$
		Calcul de $u_1 = H(m).w\,(q)$
		Calcul de $u_2 = r.w\,(q)$
		Verifier que $u_1 G + u_2 Y_A$ se

		trouve bien sur $E(\mathbb{F}_p)$
		$v = x(u_1 G + u_2 Y_A)\ (q)$
		Verifier que $v = r$

Comme on le voit, DSA et ECDSA sont presques identiques, la seule différence consiste à utiliser des points de courbes elliptiques commes clés publiques et la fonction de projection $P \to x(P)$.

EC-ELGAMAL

ElGamal

Alice	**Environnement Public**	**Bob**
Accord sur un entier premier 'très grand' p et sur un élément g de $\mathbb{Z}/p\mathbb{Z}$ d'un ordre "très grand"		Accord sur un entier premier 'très grand' p et sur un élément g de $\mathbb{Z}/p\mathbb{Z}$ d'un ordre "très grand"
Generation de Clefs		
Choix d'une clef privée $a \in [1, p-1]$		
Calcul de $A = g^a$		
	Publication de A comme clef publique	
Chiffrement d'un message m		
		Choix d'une clef 'éphémère' k.
		Calcul de : $c_1 = g^k(p)$ $c_2 = mA^k$
	$\leftarrow (c_1, c_2)$	
Réception du message chiffré $m' = (c_1, c_2)$		
Déchiffrement d'un message m'		
Calcul de		

$m = {c_1}^{-a} c_2\,(p)$		

EC-ElGamal

Alice	Environnement Public	Bob
Accord sur un entier premier 'très grand' p ,un corps $\mathbb{F}_p$, une courbe elliptique $E(\mathbb{F}_p)$ et un point $G \in E(\mathbb{F}_p)$ (d'ordre 'tres important')		Accord sur un entier premier 'très grand' p ,un corps $\mathbb{F}_p$, une courbe elliptique $E(\mathbb{F}_p)$ et un point $G \in E(\mathbb{F}_p)$ (d'ordre 'tres important')
Generation de Clefs		
Choix d'une clef privée $a \in [1, p-1]$		
Calcul de $A = aG$, un point de $E(\mathbb{F}_p)$		
	Publication de A comme clef publique	
Chiffrement d'un message m		
		Choix d'une clef 'éphémère' k.
		Calcul de : $C_1 = kG$ $C_2 = m + kA$
	$\leftarrow (C_1, C_2)$	
Réception du message chiffré $m' = (C_1, C_2)$		
Déchiffrement d'un message m'		
Calcul de $m = C_2 - a{C_1}^{-a}$		

Le chiffrement El-Gamal sur les courbes elliptiques est quasiment identique au chiffrement El-Gamal ‘classique’. Le seul problème est de représenter un message *m* comme un élément de $E(\mathbb{F}_p)$.

References

[1] An invitation to the mathematics of Fermat-Wiles, Yves Hellegouarch. Academic Press; 1st edition (October 17, 2001)

[2] Discrete Logarithms on Elliptic Curves , Aaron Blumenfel. RoseHulman Undergraduate Mathematics Journal.Volume 12, no. 1, Spring 2011

[3] An introduction to mathematical cryptography.Authors: Jeffrey Hoffstein, Jill Pipher, J.H. Silverman. Series: Undergraduate texts in mathematics. Publisher: Springer, Year: 2008

[4] Jean-Charles Faugère, Ludovic Perret, Christophe Petit, Guénaël Renault. Improving the Complexity of Index Calculus Algorithms in Elliptic Curves over Binary Fields. Eurocrypt 2012 - 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Apr 2012, Cambridge, United Kingdom. pp.27-44, ff10.1007/978-3-642-29011-4_4ff. ffhal-00776066

[5] ALGORITHMES POUR RÉSOUDRE LE PROBLÈME DU LOGARITHME DISCRET DANS LES CORPS FINIS par Antoine Joux & Reynald Lercier. In Nouvelles Méthodes Mathématiques en Cryptographie, Fascicule Journées Annuelles, pages 23–53. Société Mathématique de France, June 2007.

[6] Working Draft AMERICAN NATIONAL STANDARD X9.62-1998 Public Key Cryptography For The Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)

[7] T. SATOH AND K. ARAKI, Fermat quotients and the polynomial time discrete log algorithm for anomalous elliptic curves, preprint, 1997

[8] Elliptic Curves Number Theory and Cryptography Second Edition, LAWRENCE C. WASHINGTON, Chapman & Hall/CRC