Courbes elliptiques & cryptographie

Anna Morra, Université Bordeaux I

21 novembre 2007

Plan de l’exposé

- Cryptographie

Plan de l’exposé

- Cryptographie
- Courbes elliptiques

Plan de l’exposé

- Cryptographie
- Courbes elliptiques
- Crypto avec les courbes elliptiques

Cryptographie

Définition

Définition. *La cryptographie est l'art de chiffrer un message de façon qu'il soit inintelligible à toute autre personne que son destinataire.*

Définition

Définition. *La cryptographie est l'art de chiffrer un message de façon qu'il soit inintelligible à toute autre personne que son destinataire.*

Un **cryptosystème** est un quadruplet: $(\mathcal{P}, \mathcal{C}, f, f^{-1})$:

- $\mathcal{P} = \{\text{textes clairs (plaintext)}\}$;
- $\mathcal{C} = \{\text{textes chiffrés (ciphertext)}\}$;
- $f : \mathcal{P} \to \mathcal{C}$ (injective) fonction de chiffrement;
- $f^{-1} : f(\mathcal{P}) \to \mathcal{P}$ fonction de dechiffrement.

Petite histoire

La cryptographie a des racines très anciennes:

Petite histoire

La cryptographie a des racines très anciennes:

- Déjà Jules César utilisait une méthode de transposition (la même idée avec des variantes est après exploitée par Vigenère au seizième siècle et Hill en 1931) (ces méthodes peuvent être cassées par une analyse de fréquence).

Petite histoire

La cryptographie a des racines très anciennes:

- Déjà Jules César utilisait une méthode de transposition (la même idée avec des variantes est après exploitée par Vigenère au seizième siècle et Hill en 1931) (ces méthodes peuvent être cassées par une analyse de fréquence).
- Pendant la Seconde Guerre Mondiale l'armée allemande utilisait une machine appelée Enigma pour crypter ses informations (cette machine a été forcée par le mathematicien britannique Turing).

Cryptographie classique

- Tous les utilisateurs doivent partager une clef secrète K;

Cryptographie classique

- Tous les utilisateurs doivent partager une clef secrète K;
- cette clef secrète permet de chiffrer et dechiffer les messages;

Cryptographie classique

- Tous les utilisateurs doivent partager une clef secrète K;
- cette clef secrète permet de chiffrer et dechiffer les messages;
- donc toute personne qui peut chiffrer un message peut aussi en dechiffrer un (avec eventuellement un petit effort);

Cryptographie classique

- Tous les utilisateurs doivent partager une clef secrète K;
- cette clef secrète permet de chiffrer et dechiffer les messages;
- donc toute personne qui peut chiffrer un message peut aussi en dechiffrer un (avec eventuellement un petit effort);
- chiffrement et dechiffrement sont computationellement équivalents.

Cryptographie classique

- Tous les utilisateurs doivent partager une clef secrète K;
- cette clef secrète permet de chiffrer et dechiffer les messages;
- donc toute personne qui peut chiffrer un message peut aussi en dechiffrer un (avec eventuellement un petit effort);
- chiffrement et dechiffrement sont computationellement équivalents.

crypto classique$=$crypto symétrique$=$crypto à clef secrète.

Problèmes

La crypto classique présentait des problèmes :

- L’échange de la clef;
- la méthode devait rester secrète;
- la longueur de la clef pour obtenir un chiffre ”parfait”.

Problèmes

La crypto classique présentait des problèmes :

- L'échange de la clef;
- la méthode devait rester secrète;
- la longueur de la clef pour obtenir un chiffre "parfait".

$\Longrightarrow$ Diffie et Hellmann (1976): première méthode à clef publique.

Cryptographie à clef publique

- Les utilisateurs qui connaissent seulement la clef de chiffrement K_E ne peuvent pas dechiffrer les messages;
- la fonction de dechiffrement f^{-1} est computationellement très difficile (sauf si on connaît des information supplementaires i.e. la clef de dechiffrement K_D).

crypto à clef publique = crypto asymétrique

Avantages (crypto asymétrique)

- On n’a pas besoin d’échanger la clef secrète;

Avantages (crypto asymétrique)

- On n’a pas besoin d’échanger la clef secrète;
- dans un cryptosystème à clef publique il peut y avoir beaucoup d’utilisateurs;

Avantages (crypto asymétrique)

- On n’a pas besoin d’échanger la clef secrète;
- dans un cryptosystème à clef publique il peut y avoir beaucoup d’utilisateurs;
- on peut l’utiliser pour l’échange d’une clef secrète à utiliser dans un cryptosystème classique;

Avantages (crypto asymétrique)

- On n’a pas besoin d’échanger la clef secrète;
- dans un cryptosystème à clef publique il peut y avoir beaucoup d’utilisateurs;
- on peut l’utiliser pour l’échange d’une clef secrète à utiliser dans un cryptosystème classique;
- on peut l’utiliser aussi pour la signature digitale.

La méthode de Diffie-Hellmann

Alice et **B**ob veulent echanger une clef secrète à utiliser dans un cryptosystème classique.

La méthode de Diffie-Hellmann

Alice et **B**ob veulent echanger une clef secrète à utiliser dans un cryptosystème classique.

1. Ils choisissent ensemble un premier p assez grand et un générateur g de $(\mathbb{Z}/p\mathbb{Z})^*$. Ces informations sont *publiques*.

La méthode de Diffie-Hellmann

Alice et **B**ob veulent echanger une clef secrète à utiliser dans un cryptosystème classique.

1. Ils choisissent ensemble un premier p assez grand et un générateur g de $(\mathbb{Z}/p\mathbb{Z})^*$. Ces informations sont *publiques*.
2. Alice choisit au hasard $a \in \{2, \ldots, p-1\}$ et elle calcule $A = g^a$. Alice rend public A.

La méthode de Diffie-Hellmann

Alice et **B**ob veulent echanger une clef secrète à utiliser dans un cryptosystème classique.

1. Ils choisissent ensemble un premier p assez grand et un générateur g de $(\mathbb{Z}/p\mathbb{Z})^*$. Ces informations sont *publiques*.
2. Alice choisit au hasard $a \in \{2, \ldots, p-1\}$ et elle calcule $A = g^a$. Alice rend public A.
3. Bob choisit au hasard $b \in \{2, \ldots, p-1\}$ et il calcule $B = g^b$. Bob rend public B.

La méthode de Diffie-Hellmann

Alice et **B**ob veulent echanger une clef secrète à utiliser dans un cryptosystème classique.

1. Ils choisissent ensemble un premier p assez grand et un générateur g de $(\mathbb{Z}/p\mathbb{Z})^*$. Ces informations sont *publiques*.
2. Alice choisit au hasard $a \in \{2, \ldots, p-1\}$ et elle calcule $A = g^a$. Alice rend public A.
3. Bob choisit au hasard $b \in \{2, \ldots, p-1\}$ et il calcule $B = g^b$. Bob rend public B.
4. Tous les deux calculent $C = g^{ab} = (g^a)^b = (g^b)^a$, qui sera leur clef secrète.

Remarques

Dans la communication on a transmis seulement A et B.
Si un intrus,**c**harles veut connaître C, il devrait résoudre le *problème de Diffie-Hellmann*:

Remarques

Dans la communication on a transmis seulement A et B.
Si un intrus,**c**harles veut connaître C, il devrait résoudre le *problème de Diffie-Hellmann*:

p premier, g générateur de $(\mathbb{Z}/p\mathbb{Z})^*$, g^a, g^b donnés,

$\Rightarrow$ calculer g^{ab}

Remarques

Dans la communication on a transmis seulement A et B.
Si un intrus,**c**harles veut connaître C, il devrait résoudre le *problème de Diffie-Hellmann*:

p premier, g générateur de $(\mathbb{Z}/p\mathbb{Z})^*$, g^a, g^b donnés,
$\Rightarrow$ calculer g^{ab}

On conjecture que ce problème est équivalent au problème du logarithme discret:

Remarques

Dans la communication on a transmis seulement A et B.
Si un intrus,**c**harles veut connaître C, il devrait résoudre le *problème de Diffie-Hellmann*:

p premier, g générateur de $(\mathbb{Z}/p\mathbb{Z})^*$, g^a, g^b donnés,
$\Rightarrow$ calculer g^{ab}

On conjecture que ce problème est équivalent au problème du logarithme discret:

p premier, g génerateur de $(\mathbb{Z}/p\mathbb{Z})^*$, $a = g^x$ donnés,
$\Rightarrow$ calculer x

Remarques

Dans la communication on a transmis seulement A et B.
Si un intrus,**c**harles veut connaître C, il devrait résoudre le *problème de Diffie-Hellmann*:

p premier, g générateur de $(\mathbb{Z}/p\mathbb{Z})^*$, g^a, g^b donnés,
$\Rightarrow$ calculer g^{ab}

On conjecture que ce problème est équivalent au problème du logarithme discret:

p premier, g génerateur de $(\mathbb{Z}/p\mathbb{Z})^*$, $a = g^x$ donnés,
$\Rightarrow$ calculer x

Pour ce dernier problème on ne connait pas d'algorithme polynomial, mais au plus sous-exponentiel (*index-calculus*).

L’algorithme du double cadenas

- Alice et Bob veulent communiquer secrètement.

L’algorithme du double cadenas

- Alice et Bob veulent communiquer secrètement.
- On suppose que chacun d’eux possède un cadenas infrangible et inviolable, que seulement le proprietaire peut ouvrir.

L’algorithme du double cadenas

- Alice et Bob veulent communiquer secrètement.
- On suppose que chacun d’eux possède un cadenas infrangible et inviolable, que seulement le proprietaire peut ouvrir.
- On suppose aussi que les deux cadenas possèdent la *propriété commutative* : on peut les appliquer/enlever en n’importe quel ordre.

1. Alice veut envoyer le message M a Bob. Elle le ferme avec son cadenas A, en obtenant $A(M)$, qu'elle envoie a Bob.

1. Alice veut envoyer le message M a Bob. Elle le ferme avec son cadenas A, en obtenant $A(M)$, qu'elle envoie a Bob.
2. Bob ne peut pas ouvrir le cadenas A, donc il ne fait que rajouter dessus son cadenas, donc il obtient $B(A(M))$ et il le renvoie à Alice.

1. Alice veut envoyer le message M a Bob. Elle le ferme avec son cadenas A, en obtenant $A(M)$, qu'elle envoie a Bob.
2. Bob ne peut pas ouvrir le cadenas A, donc il ne fait que rajouter dessus son cadenas, donc il obtient $B(A(M))$ et il le renvoie à Alice.
3. Alice ouvre son cadenas en obtenant $B(M)$ qu'elle renvoie à Bob.

1. Alice veut envoyer le message M a Bob. Elle le ferme avec son cadenas A, en obtenant $A(M)$, qu'elle envoie a Bob.
2. Bob ne peut pas ouvrir le cadenas A, donc il ne fait que rajouter dessus son cadenas, donc il obtient $B(A(M))$ et il le renvoie à Alice.
3. Alice ouvre son cadenas en obtenant $B(M)$ qu'elle renvoie à Bob.
4. Bob ouvre le cadenas B et peut enfin lire le message M.

Cryptosystème de Massey-Omura

- Tous les utilisateurs s'accordent sur un premier p assez grand.

Cryptosystème de Massey-Omura

- Tous les utilisateurs s'accordent sur un premier p assez grand.
- Chaque utilisateur choisit un élément $e \in (\mathbb{Z}/(p-1)\mathbb{Z})^*$ et il calcule $d \equiv e^{-1}(\mod (p-1))$.

Cryptosystème de Massey-Omura

- Tous les utilisateurs s'accordent sur un premier p assez grand.
- Chaque utilisateur choisit un élément $e \in (\mathbb{Z}/(p-1)\mathbb{Z})^*$ et il calcule $d \equiv e^{-1}(\mod (p-1))$.
- e et d sont secrets.

Cryptosystème de Massey-Omura

- Tous les utilisateurs s'accordent sur un premier p assez grand.
- Chaque utilisateur choisit un élément $e \in (\mathbb{Z}/(p-1)\mathbb{Z})^*$ et il calcule $d \equiv e^{-1}(\mod (p-1))$.
- e et d sont secrets.
- Or, si Alice veut envoyer le message $M \in \mathbb{Z}/p\mathbb{Z}$ à Bob, ils agissent de la manière suivante:

Algorithme de Massey-Omura

1. Alice calcule $A = M^{e_A} \pmod{p}$ et l'envoie à Bob.

Algorithme de Massey-Omura

1. Alice calcule $A = M^{e_A} \pmod{p}$ et l'envoie à Bob.
2. Bob calcule $B = A^{e_B} = M^{e_A e_B} \pmod{p}$ et le renvoie à Alice.

Algorithme de Massey-Omura

1. Alice calcule $A = M^{e_A} \pmod{p}$ et l'envoie à Bob.
2. Bob calcule $B = A^{e_B} = M^{e_A e_B} \pmod{p}$ et le renvoie à Alice.
3. Alice calcule $C = B^{d_A} = M^{e_A e_B d_A} = M^{e_A e_B e_A^{-1}} = M^{e_B} \pmod{p}$ et l'envoie à Bob (Alice a enlevé son cadenas).

Algorithme de Massey-Omura

1. Alice calcule $A = M^{e_A} \pmod{p}$ et l'envoie à Bob.
2. Bob calcule $B = A^{e_B} = M^{e_A e_B} \pmod{p}$ et le renvoie à Alice.
3. Alice calcule $C = B^{d_A} = M^{e_A e_B d_A} = M^{e_A e_B e_A^{-1}} = M^{e_B} \pmod{p}$ et l'envoie à Bob (Alice a enlevé son cadenas).
4. Bob calcule $D = C^{d_B} = C^{e_B^{-1}} = M^{e_B e_B^{-1}} = M \pmod{p}$.

Remarques

L’algorithme de Massey-Omura a les mêmes avantages et les mêmes défauts de l’algorithme du double cadenas.

Remarques

L'algorithme de Massey-Omura a les mêmes avantages et les mêmes défauts de l'algorithme du double cadenas.

Le problème principal est qu'on a besoin d'une méthode d'authentification (signature numérique) pour eviter les intrusions.

Cryptosystème de ElGamal

- Ce cryptosystème utilise le problème du logarithme discret.

Cryptosystème de ElGamal

- Ce cryptosystème utilise le problème du logarithme discret.
- Tous les utilisateurs s'accordent sur un premier p assez grand et un générateur g de $(\mathbb{Z}/p\mathbb{Z})^*$.

Cryptosystème de ElGamal

- Ce cryptosystème utilise le problème du logarithme discret.
- Tous les utilisateurs s'accordent sur un premier p assez grand et un générateur g de $(\mathbb{Z}/p\mathbb{Z})^*$.
- Chaque utilisateur choisit sa clef privée $x \in (\mathbb{Z}/p\mathbb{Z})^*$ et rend public g^x.

Algorithme de ElGamal

Alice: clef privée x, clef publique $a = g^x$;
Bob : clef privée y, clef publique $b = g^y$.
Alice veut envoyer à Bob le message M.

Algorithme de ElGamal

Alice: clef privée x, clef publique $a = g^x$;
Bob : clef privée y, clef publique $b = g^y$.
Alice veut envoyer à Bob le message M.

1. Alice choisit au hasard un élément $k \in (\mathbb{Z}/p\mathbb{Z})^*$;

Algorithme de ElGamal

Alice: clef privée x, clef publique $a = g^x$;
Bob : clef privée y, clef publique $b = g^y$.
Alice veut envoyer à Bob le message M.

1. Alice choisit au hasard un élément $k \in (\mathbb{Z}/p\mathbb{Z})^*$;
2. Alice envoye à Bob le couple (g^k, Mb^k);

Algorithme de ElGamal

Alice: clef privée x, clef publique $a = g^x$;
Bob : clef privée y, clef publique $b = g^y$.
Alice veut envoyer à Bob le message M.

1. Alice choisit au hasard un élément $k \in (\mathbb{Z}/p\mathbb{Z})^*$;
2. Alice envoye à Bob le couple (g^k, Mb^k);
3. Bob calcule $g^{ky} = b^k$ donc il retrouve $M = Mb^k g^{-ky}$.

Courbes elliptiques

Définition

Définition. *Une courbe elliptique E sur un corps K est une courbe non singulière de degré 3 dans le plan projectif, constituée par les couples $(X, Y) \in K^2$ qui satisfont l'équation affine*

$$Y^2 + a_1XY + a_3Y = X^3 + a_2X^2 + a_4X + a_6, \quad a_i \in K$$

plus un point à l'infini O.
On note $E(K)$ l'ensemble des points de cette courbe.

Définition

Définition. *Une courbe elliptique E sur un corps K est une courbe non singulière de degré 3 dans le plan projectif, constituée par les couples $(X, Y) \in K^2$ qui satisfont l'équation affine*

$$Y^2 + a_1XY + a_3Y = X^3 + a_2X^2 + a_4X + a_6, \quad a_i \in K$$

plus un point à l'infini O.
On note $E(K)$ l'ensemble des points de cette courbe.

Définition. *Si $\mathrm{car}(K) \neq 2, 3$ l'equation générale d'une courbe elliptique est*

$$Y^2 = X^3 + aX + b$$

et la condition de non singularité devient

$$4a^3 + 27b^2 \neq 0$$

La loi de groupe

On donne à E une structure de groupe telle que:

La loi de groupe

On donne à E une structure de groupe telle que:

1. O est l'élément neutre du groupe;

La loi de groupe

On donne à E une structure de groupe telle que:

1. O est l'élément neutre du groupe;
2. si $P \neq O$ alors $-P$ est l'unique autre point de la courbe ($\neq O$) qui a la même abscisse que P;

La loi de groupe

On donne à E une structure de groupe telle que:

1. O est l'élément neutre du groupe;
2. si $P \neq O$ alors $-P$ est l'unique autre point de la courbe ($\neq O$) qui a la même abscisse que P;
3. si $P = -Q$ alors $P + Q = O$;

La loi de groupe

On donne à E une structure de groupe telle que:

1. O est l'élément neutre du groupe;
2. si $P \neq O$ alors $-P$ est l'unique autre point de la courbe ($\neq O$) qui a la même abscisse que P;
3. si $P = -Q$ alors $P + Q = O$;
4. si $P \neq \pm Q$ alors pour obtenir $P + Q$ il faut tracer la droite ℓ qui passe par P et Q, trouver le troisième point d'intersection de ℓ avec E, disons R, et définir $P + Q = -R$;

La loi de groupe

On donne à E une structure de groupe telle que:

1. O est l'élément neutre du groupe;
2. si $P \neq O$ alors $-P$ est l'unique autre point de la courbe ($\neq O$) qui a la même abscisse que P;
3. si $P = -Q$ alors $P + Q = O$;
4. si $P \neq \pm Q$ alors pour obtenir $P + Q$ il faut tracer la droite ℓ qui passe par P et Q, trouver le troisième point d'intersection de ℓ avec E, disons R, et définir $P + Q = -R$;
5. si $P = Q$ alors on fait comme au point 3, sauf que la droite ℓ est la tangente à E au point P.

Formules explicites

Supposons $\mathrm{car}(K) \neq 2, 3$. $P = (x_1, y_1)$, $Q = (x_2, y_2)$.
Alors $-P = (x_1, -y_1)$ et $P + Q = (x_3, y_3)$ où:

- Si $P \neq \pm Q$:

$$\begin{cases} x_3 = \left(\frac{y_2 - y1}{x_2 - x_1}\right)^2 - x_1 - x_2 \\ y_3 = -y_1 + \frac{(y_2 - y_1)(x1 - x_3)}{x_2 - x_1} \end{cases}$$

- Si $P = Q$

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

Multiples des points

Définition. *Soit P un point sur une courbe elliptique E. Soit $k \in \mathbb{Z}$. On définit kP de la manière suivante:*

- $kP = O$ *si* $k = 0$;
- $kP = P + P + \cdots + P$ *(k fois) si* $k > 0$;
- $kP = (-P) + (-P) + \cdots + (-P)$ *($-k$ fois) si* $k < 0$.

Courbes elliptiques sur $\mathbb{F}_q$

Soit $K = \mathbb{F}_q$ ($q = p^r$) un corps fini.
Soit E une courbe elliptique définie sur K.
La courbe E aura donc un nombre fini de points sur le corps K. Plus précisement:

Théorème (Hasse)**.** *Soit E une courbe elliptique sur $\mathbb{F}_q$. Alors*

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

Crypto avec les courbes elliptiques

ECDLP

(ECDLP=Elliptic Curve Discrete Logarithm Problem)

Soient E une courbe elliptique sur $K = \mathbb{F}_q$, $q = p^r$, B un point de E.

ECDLP

(ECDLP=Elliptic Curve Discrete Logarithm Problem)

Soient E une courbe elliptique sur $K = \mathbb{F}_q$, $q = p^r$, B un point de E.

Le problème du logarithme discret sur la courbe elliptique E en base B est :

ECDLP

(ECDLP=Elliptic Curve Discrete Logarithm Problem)

Soient E une courbe elliptique sur $K = \mathbb{F}_q$, $q = p^r$, B un point de E.

Le problème du logarithme discret sur la courbe elliptique E en base B est :

étant donné $P \in E(K)$, trouver, si il existe, $x \in \mathbb{Z}$ tel que $xB = P$.

Analogue de Diffie-Hellmann

Alice et Bob veulent s’accorder sur une clef secrète à utiliser dans un cryptosystème symetrique.

Analogue de Diffie-Hellmann

Alice et Bob veulent s'accorder sur une clef secrète à utiliser dans un cryptosystème symetrique.

1. Ils fixent (publiquement) un corps $\mathbb{F}_q$, $q = p^r$ et une courbe elliptique E sur ce corps.

Analogue de Diffie-Hellmann

Alice et Bob veulent s'accorder sur une clef secrète à utiliser dans un cryptosystème symetrique.

1. Ils fixent (publiquement) un corps $\mathbb{F}_q$, $q = p^r$ et une courbe elliptique E sur ce corps.
2. Ils rendent public un point $B \in E$ (base) qui ait un ordre m assez grand.

Analogue de Diffie-Hellmann

Alice et Bob veulent s'accorder sur une clef secrète à utiliser dans un cryptosystème symetrique.

1. Ils fixent (publiquement) un corps $\mathbb{F}_q$, $q = p^r$ et une courbe elliptique E sur ce corps.
2. Ils rendent public un point $B \in E$ (base) qui ait un ordre m assez grand.
3. Alice choisit un entier a ($\sim m$) secret, et envoie (publiquement) à Bob aB.

Analogue de Diffie-Hellmann

Alice et Bob veulent s'accorder sur une clef secrète à utiliser dans un cryptosystème symetrique.

1. Ils fixent (publiquement) un corps $\mathbb{F}_q$, $q = p^r$ et une courbe elliptique E sur ce corps.
2. Ils rendent public un point $B \in E$ (base) qui ait un ordre m assez grand.
3. Alice choisit un entier a ($\sim m$) secret, et envoie (publiquement) à Bob aB.
4. Bob choisit un entier b ($\sim m$) et envoie à Alice bB.

Analogue de Diffie-Hellmann

Alice et Bob veulent s'accorder sur une clef secrète à utiliser dans un cryptosystème symetrique.

1. Ils fixent (publiquement) un corps $\mathbb{F}_q$, $q = p^r$ et une courbe elliptique E sur ce corps.
2. Ils rendent public un point $B \in E$ (base) qui ait un ordre m assez grand.
3. Alice choisit un entier a ($\sim m$) secret, et envoie (publiquement) à Bob aB.
4. Bob choisit un entier b ($\sim m$) et envoie à Alice bB.
5. Tous les deux calculent la clef secrète abB.

Analogue de Massey-Omura

- On fixe une courbe elliptique E sur un corps $\mathbb{F}_q$ (q assez grand).
- On calcule $N =$ le nombre des points de la courbe.
- Chaque utilisateur choisit un entier au hasard, e entre 1 et N tel que $\mathrm{pgcd}(e, N) = 1$ et il calcule $d \equiv e^{-1}(\mod N)$.
- On convertit les messages en points de la courbe elliptique E, de façon à transmettre des points de la courbe au lieu du message original.

Si Alice veut envoyer à Bob le message $M \in E(K)$:

Si Alice veut envoyer à Bob le message $M \in E(K)$:

1. Alice envoie $e_A M$ à Bob.

Si Alice veut envoyer à Bob le message $M \in E(K)$:

1. Alice envoie $e_A M$ à Bob.
2. Bob envoie $e_B(e_A M)$ à Alice.

Si Alice veut envoyer à Bob le message $M \in E(K)$:

1. Alice envoie $e_A M$ à Bob.
2. Bob envoie $e_B(e_A M)$ à Alice.
3. Alice envoie à Bob $d_A(e_B e_A M) = e_B M$.

Si Alice veut envoyer à Bob le message $M \in E(K)$:

1. Alice envoie $e_A M$ à Bob.
2. Bob envoie $e_B(e_A M)$ à Alice.
3. Alice envoie à Bob $d_A(e_B e_A M) = e_B M$.
4. Bob peut lire le message $M = d_B(e_B M)$.

Analogue de ElGamal

On fixe une courbe elliptique E sur un corps $\mathbb{F}_q$ et une base $B \in E$. Chaque utilisateur choisit un entier au hasard a qui sera sa clef secrète, et il rend public aB.
Si Alice veut envoyer à Bob le message $M \in E(K)$:

1. Alice choisit un entier au hasard k et envoie à Bob $(kB, M + k(a_B B))$
2. Bob peut retrouver le message original

$$M = M + k(a_B B) - a_B(kB).$$

Avantages

- On peut construire plusieurs groupes abeliens (points d’une courbe elliptique) sur le même corps fini.

Avantages

- On peut construire plusieurs groupes abeliens (points d'une courbe elliptique) sur le même corps fini.
- Il n'existe pas d'algorithme sous-exponentiel pour résoudre le problème du logarithme discret sur les courbes elliptiques (non supersingulières)

Avantages

- On peut construire plusieurs groupes abeliens (points d'une courbe elliptique) sur le même corps fini.
- Il n'existe pas d'algorithme sous-exponentiel pour résoudre le problème du logarithme discret sur les courbes elliptiques (non supersingulières)
- on peut utiliser des clefs plus courtes pour obtenir la même securité.

Merci!