Fondements mathématiques des courbes elliptiques et application au système de chiffrement ElGamal

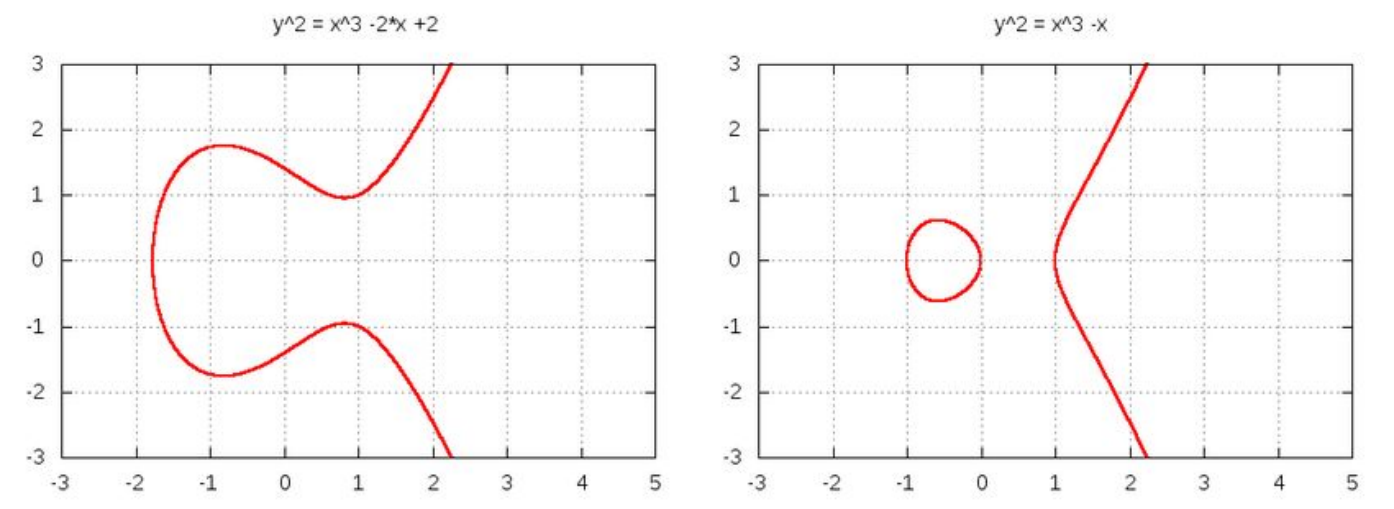
DESNEUX Raphaël
GEA Nolan

Qu’est-ce qu’une courbe elliptique ?

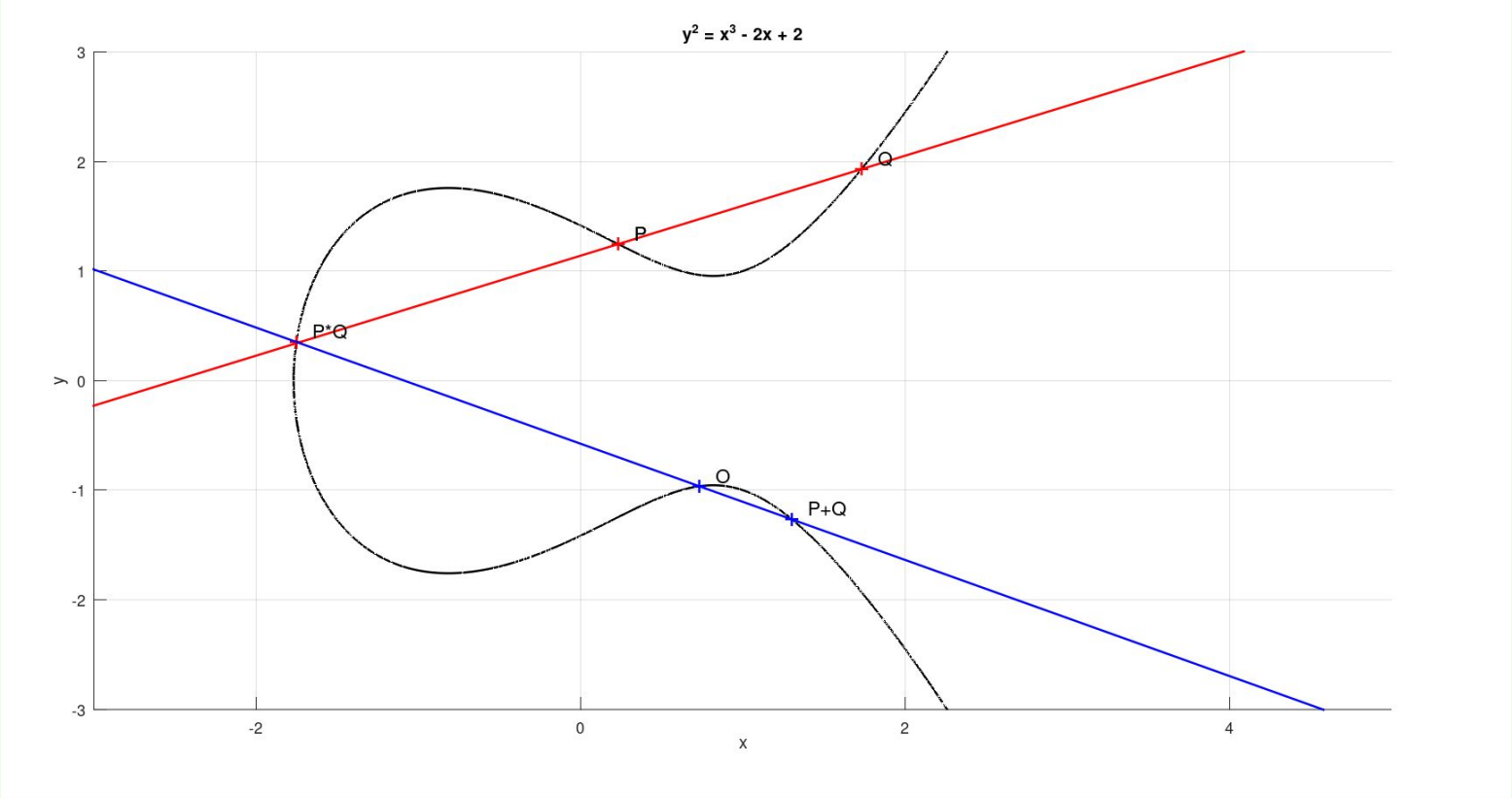
Ensemble des points de coordonnées (x,y) vérifiant l’équation (non singulière) :

$$ax^3 + bx^2y + cxy^2 + dy^3 + ex^2 + fxy + gy^2 + hx + iy + j = 0$$

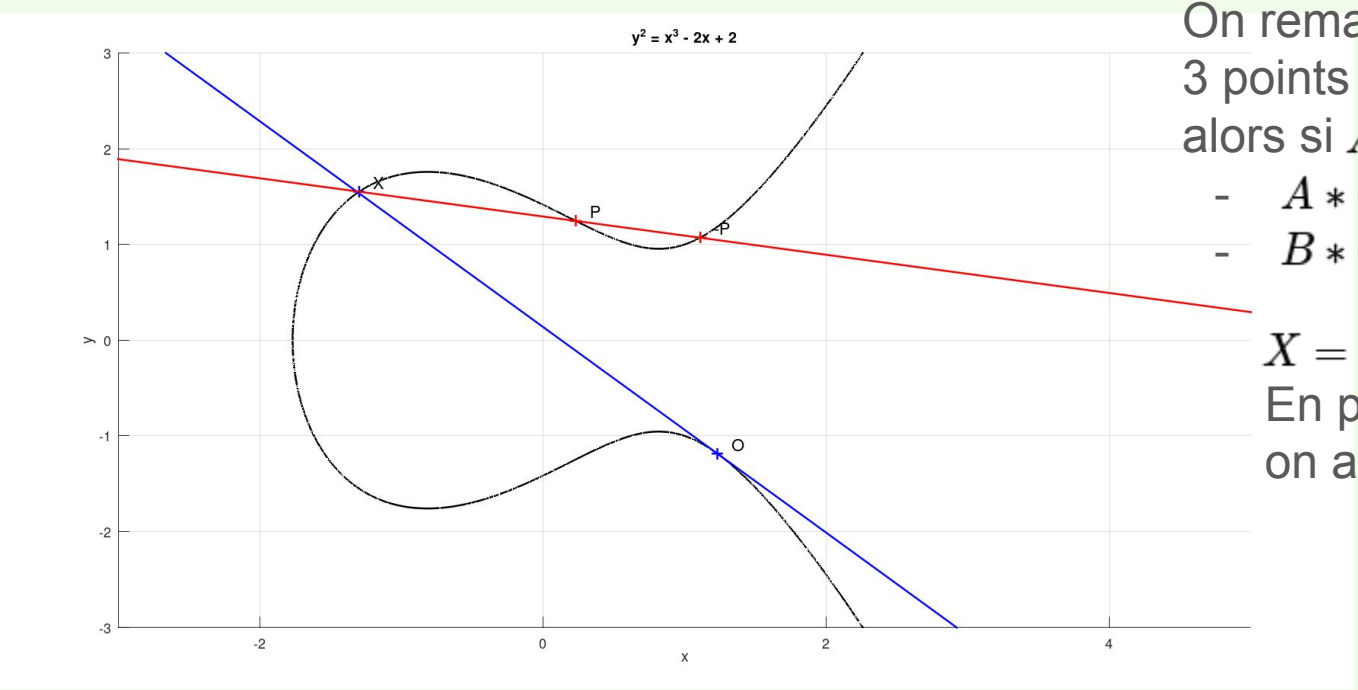
où $a, b, c, d, e, f, g, h, i, j \in \mathbb{R}, x, y \in \mathbb{K}$ où $\mathbb{K}$ est un corps



Construction géométrique de la loi de groupe (commutative)



Existence de l'opposé (ou de l'inverse)



On remarque que si A, B, C sont 3 points de la courbe elliptique alors si $A * B = C$ on a :

- $A * C = B$
- $B * C = A$

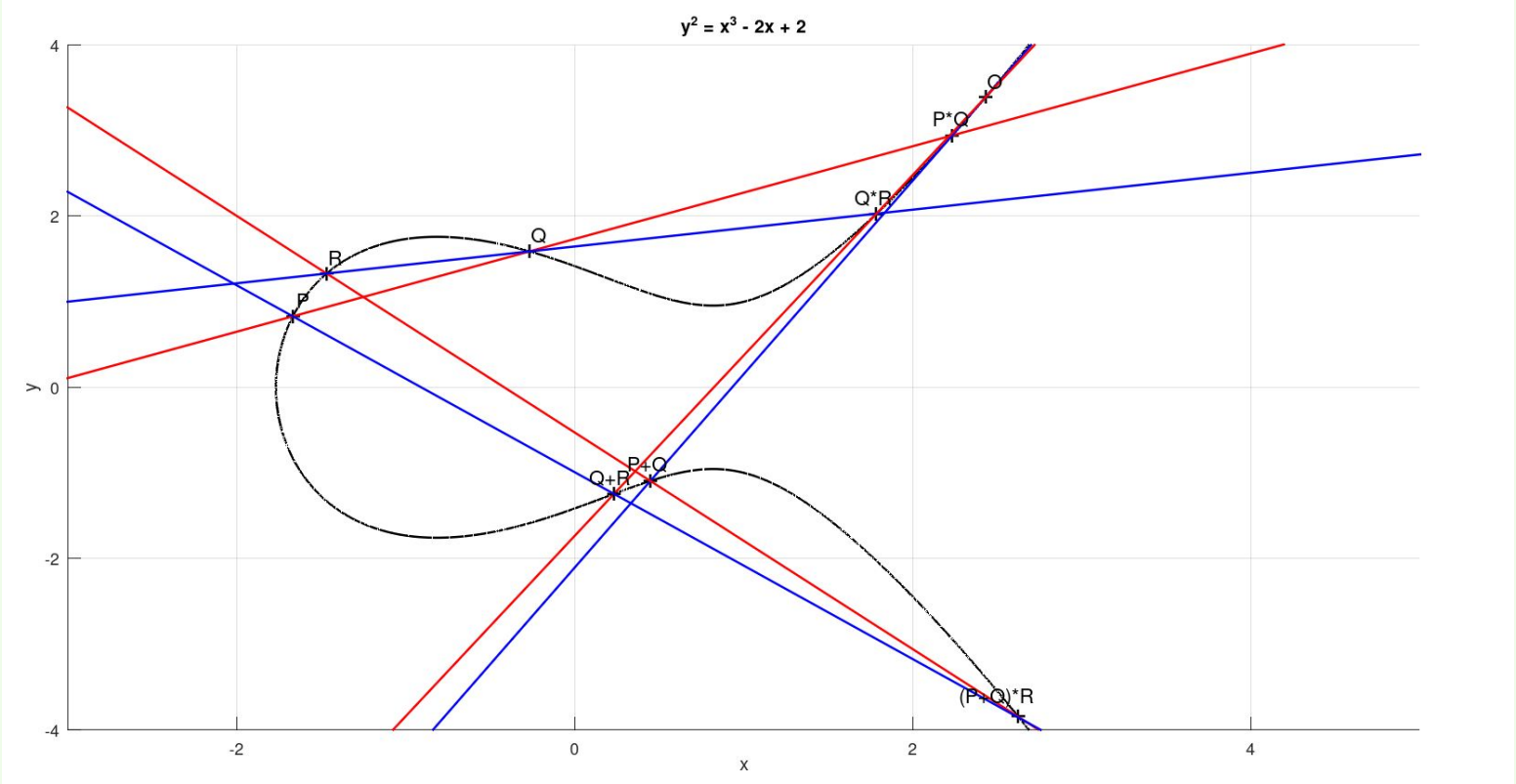
$X = O * O$

En posant $-P = P * X$

on a $-P + P = ((-P) * P) * O$

$= X * O = O$

Associativité



Calcul de P*Q

Forme de Weierstrass :

Les équations des courbes elliptiques définies sur $\mathbb{R}$ (ou sur $\mathbb{Q}$) peuvent s'écrire sous la forme :

$y^2 = x^3 + ax + b$ où $a, b \in \mathbb{R}$ (ou $\mathbb{Q}$)

En étudiant des égalités de polynôme, on obtient ces formules :

en notant on a $P = (x_1, y_1), Q = (x_2, y_2), P * Q = (x_3, y_3)$ $\begin{cases} x_3 = \lambda^2 - x_1 - x_2 \\ y_3 = \lambda x + \nu \end{cases}$

où $\lambda = \dfrac{y_1 - y_2}{x_1 - x_2}$ et $\nu = y_1 - \lambda x_1$

Plan projectif

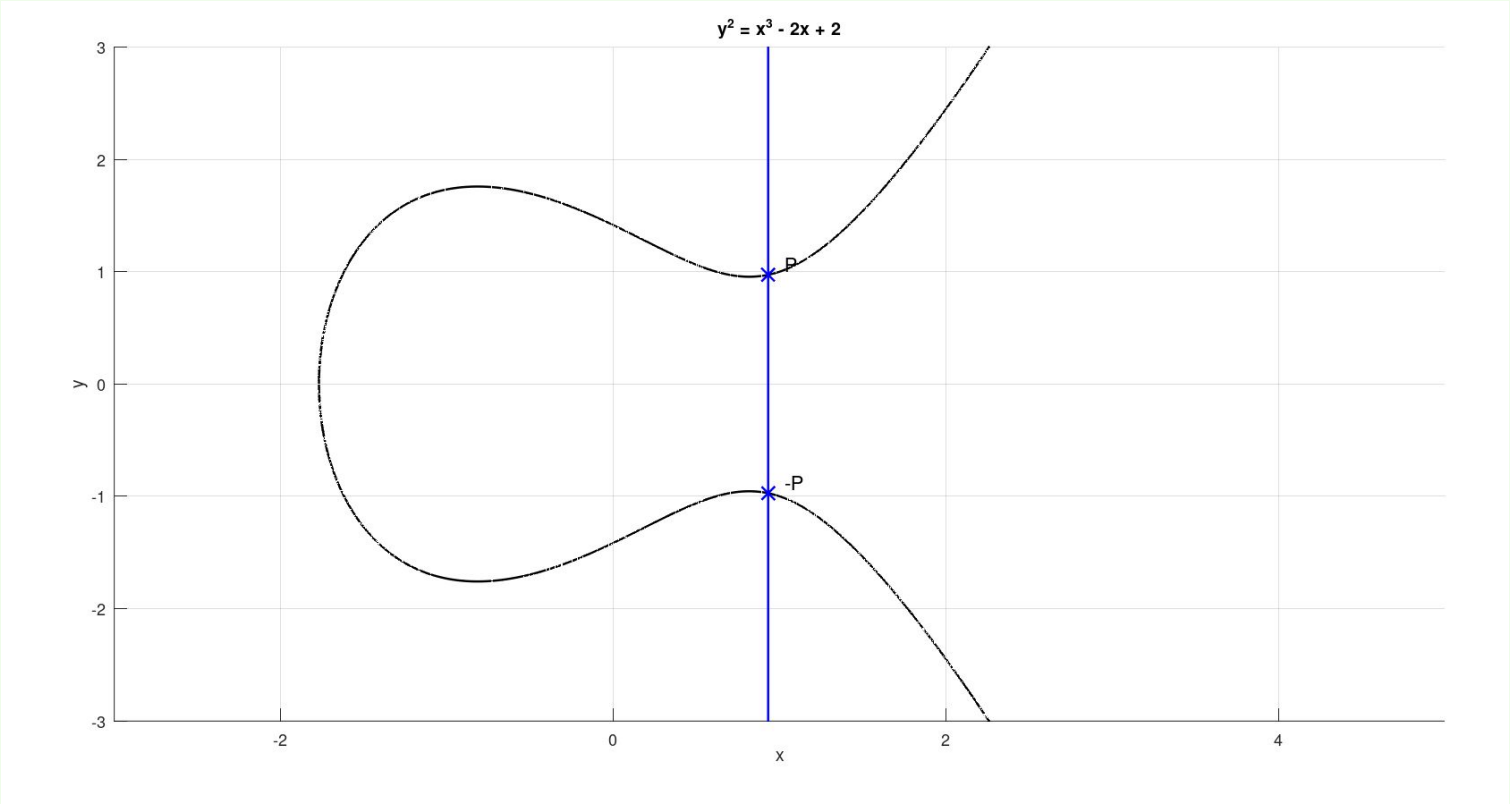
Définition : Espace de dimension 2 dans lequel on ajoute un point à l'infini.

On choisit désormais le point à l'infini comme neutre du groupe $(E(\mathbb{K}), +)$.
Par définition de la courbe, il est infiniment en haut et en bas

Nouvelles propriétés :

- Si $P = (x, y)$ est un point (non à l'infini), $P * O = (x, -y)$
- Si P et Q sont des points tels que $P * Q = (x', y')$
  alors $P + Q = (P * Q) * O = (x', -y')$
- $O * O = O$
- Si $P = (x, y)$, alors $-P = (x, -y)$

Plan projectif



Courbes elliptiques définies sur un groupe cyclique

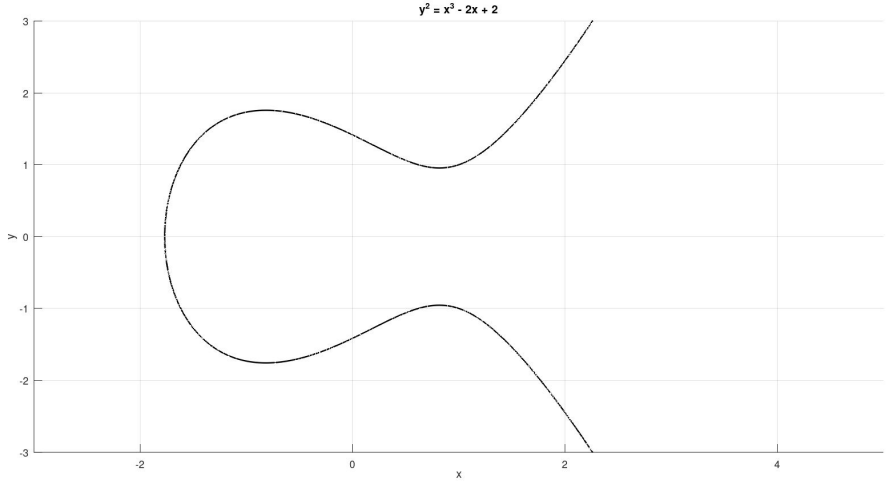
Définition : Un groupe cyclique est un groupe à la fois monogène et fini.

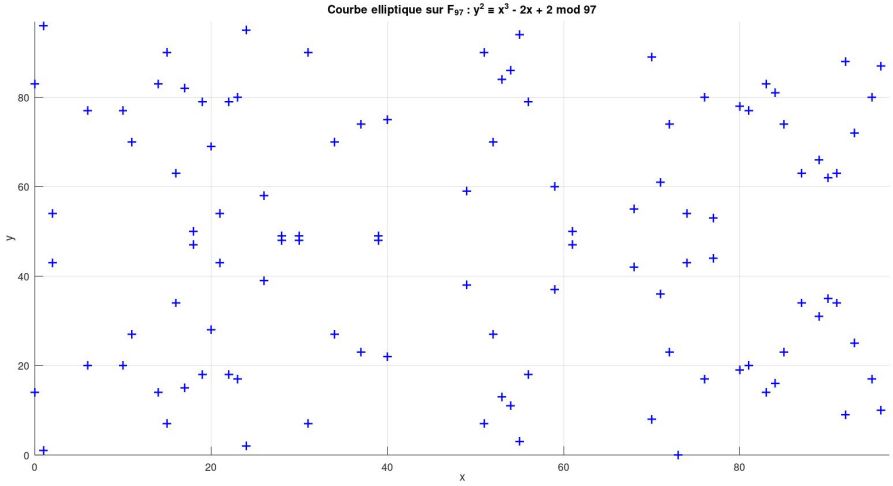
Pour tout groupe cyclique $\mathbb{G}$, il existe $n \in \mathbb{N}$ premier tel que $\mathbb{G} \sim \mathbb{Z}/n\mathbb{Z}$
où $\mathbb{Z}/n\mathbb{Z} = \{k \mod n | k \in \mathbb{Z}\}$

Application aux courbes elliptiques : Pour chaque couple d'entier (x,y) vérifiant l'équation de la courbe, on applique un modulo à x et à y

L'équation de la courbe devient donc : $y^2 \equiv x^3 + ax + b \mod n$

Courbes elliptiques définies sur un groupe cyclique





Implémentation en Python des courbes

Création des objets :

- Courbe elliptique doté notamment des attributs f, fp et p tels que l'équation de la courbe soit $y^2 \equiv f(x) \bmod p$,et fp = f'
- Point doté notamment des attributs courbe (courbe elliptique à laquelle il appartient) et x et y (coordonnées dans le plan)
- Infini (qui hérite de point) qui représente le point à l'infini

*ÉTOILE (appelé par la commande P*Q)*

ENTRÉE : points P et Q de la même courbe
DÉBUT
    SI P.x = Q.x **FAIRE**
        SI P.y != Q.y ou P.y = 0 **FAIRE**
            RENVOYER Infini
        FIN SI
        lambda ← P.courbe.fp(P.x) * inv_mod(2*P.y, P.courbe.p)
    SINON FAIRE
        lambda ← (P.y - Q.y) * inv_mod(P.x - Q.x, P.courbe.p)
    FIN SI
    x3 ← lambda**2 - P.x - Q.x
    y3 ← lambda * x3 + (P.y - lamb * P.x)
    RENVOYER Point(x = x3, y = y3, courbe = P.courbe)
FIN

*MUL_BY_INT (appelé par la commande a*P ou P*a)*

ENTRÉE : un point P et un entier a
DÉBUT :
    SI a < 0 **FAIRE**
        RENVOYER (-a) * (-P)
    FIN SI
    SI a = 0 **FAIRE**
        RENVOYER Infini
    FIN SI
    sub ← P * (a // 2)
    SI a % 2 = 1 **FAIRE**
        RENVOYER sub + sub + P
    FIN SI
    RENVOYER sub + sub
FIN

Problème du logarithme discret

Logarithme discret : fonction réciproque de l'exponentiation discrète : $k \mapsto b^k \mod n$
Soient $a, b \in \mathbb{Z}/n\mathbb{Z}, k \in \mathbb{N}$ tels que k est le plus petit entier naturel vérifiant $a = b^k$,
k est donc appelé le logarithme discret en base b. $0 \leq k < n$

Exemple : Dans $\mathbb{Z}/7\mathbb{Z}$:
$\log_3(6) = 3$ car $3^3 \mod 7 = 27 \mod 7 = 6$

Propriété du logarithme discret : Il n'existe pas de moyen de le calculer rapidement et de manière sûre. On va utiliser cette propriété dans le fonctionnement d'ElGamal.

Principe général du cryptosystème ElGamal

On suppose que Bob veut envoyer un message à Alice

Alice choisit :

- Une clé secrète $s \in \mathbb{N}$

Elle partage (clé publique) :

- Un groupe G où le problème du logarithme discret est difficile
- Un élément de $P \in G$ générateur ou d'ordre suffisamment grand
- $B = P^s$ (notation multiplicative), $B = sP$ (notation additive)

Principe général du cryptosystème ElGamal

Bob choisit :

- un élément $M \in G$ (message)
- une autre clé secrète $k \in \mathbb{N}$

Il partage (message encrypté) :

- $M_1 = P^k$ (notation multiplicative), $M_1 = kP$ (notation additive)
- $M_2 = M \times B^k$ (notation multiplicative), $M_2 = M + kB$ (notation additive)

Alice a donc juste à calculer : $M_2 \times (M_1^s)^{-1} = M \times P^{sk} \times P^{-sk} = M$ (notation multiplicative)

$M_2 - sM_1 = M + skP - skP = M$ (notation additive)

Les courbes elliptiques avec ElGamal

Adaptation d'ElGamal aux courbes elliptiques :

- Le groupe G est l'ensemble des points d'une courbe elliptique modulo un grand nombre premier, représenté par la courbe elliptique en elle-même
- Le point P est un point de G

GÉNÉRER_CLE_PUBLIQUE

ENTRÉE : la clé secrète d'Alice s, CE un courbe elliptique (modulaire) et P un point de CE (d'ordre grand)
DÉBUT
    B ← s * P
    RENVOYER (CE, P, B)
FIN

CHIFFRAGE

ENTRÉE : la clé publique d'Alice (CE, P, B) et le message M de Bob (un de CE)
DÉBUT
    k ← nombre aléatoire entre 1 et CE.p - 1
    M1 ← k*P
    M2 ← M + k*B
    RENVOYER (M1, M2)
FIN

DÉCHIFFRAGE

ENTRÉE : la clé secrète s d'Alice et le message chiffré (M1, M2) de Bob
DÉBUT
    M ← M2 - s*M1
    RENVOYER M
FIN

Les progrès permis par l'utilisation des courbes elliptiques avec ElGamal

- Problème du logarithme discret plus compliqué sur les courbes elliptiques
- Taille de l'ordre du groupe réduite pour une sécurité équivalente
- Taille du message encrypté également réduite
- Calculs plus rapides
- Plus faible consommation d'énergie

En clair, EC-ElGamal constitue un système de chiffrement léger mais néanmoins efficace. Cela explique son utilisation dans des systèmes limités tels que les systèmes embarqués.