Cryptanalyse

Cours 12 - Applications de la Réduction de Réseau en Cryptanalyse

Maxime Bombar

Mardi 26 Novembre

Rappel : Examen Final

Examen final de cryptanalyse : Mardi 17 Décembre 14h30

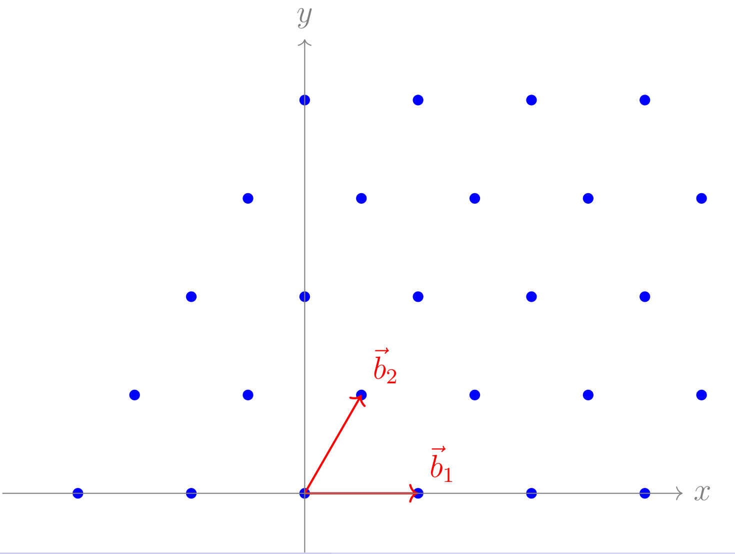
Documents autorisés : une feuille A4 manuscrite (recto-verso)

Rappels de la Semaine Dernière

Réseau Euclidien

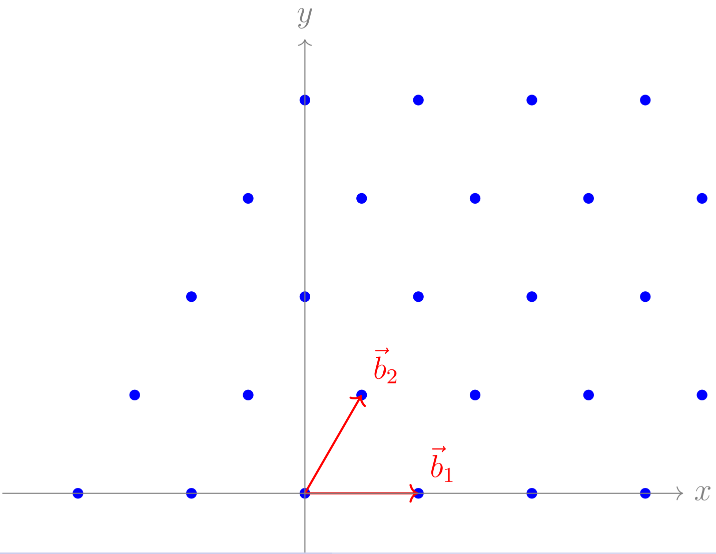
- Un réseau Euclidien est un sous-groupe discret de $\mathbb{R}^n$, muni de la norme Euclidienne $||\cdot||_2$.
- De manière équivalente, c'est l'ensemble des combinaisons linéaires **entières** d'une famille libre de $\mathbb{R}^n$:

$$\Lambda \stackrel{\text{def}}{=} \left\{ \sum_i x_i b_i \mid x_i \in \mathbb{Z} \right\}.$$



Base d'un Réseau

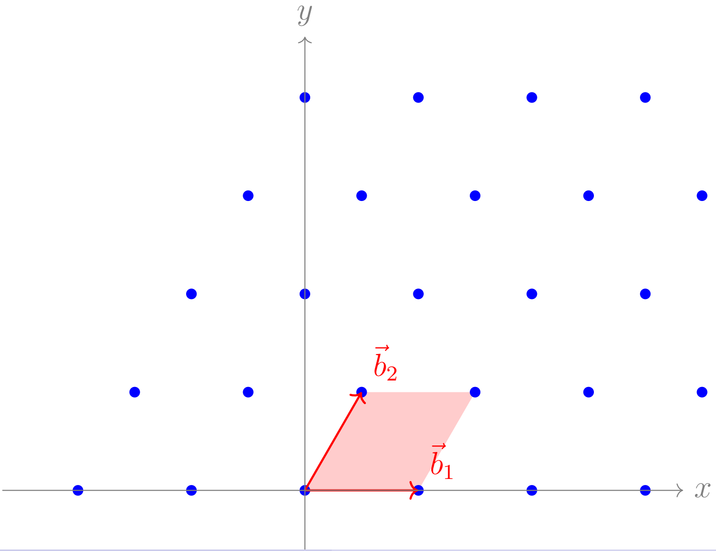
- Toutes les bases d'un réseau ont le même cardinal, appelé rang du réseau.
- En dimension n, un réseau est de rang au plus n.



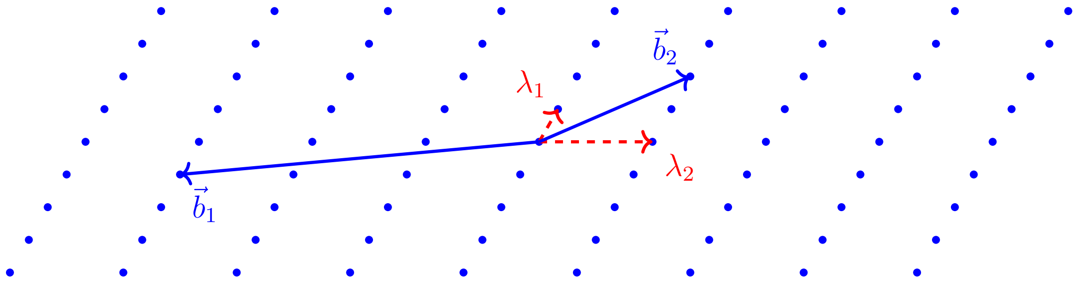
Un Invariant Important : Déterminant

- Le volume défini par les vecteurs de base est un invariant du réseau. On l'appelle déterminant du réseau.

- Dans le cas d'un réseau de rang plein, on peut représenter les vecteurs de base dans une matrice $B \in \mathbb{R}^{n \times n}$. Le déterminant est alors $|\det(B)|$.

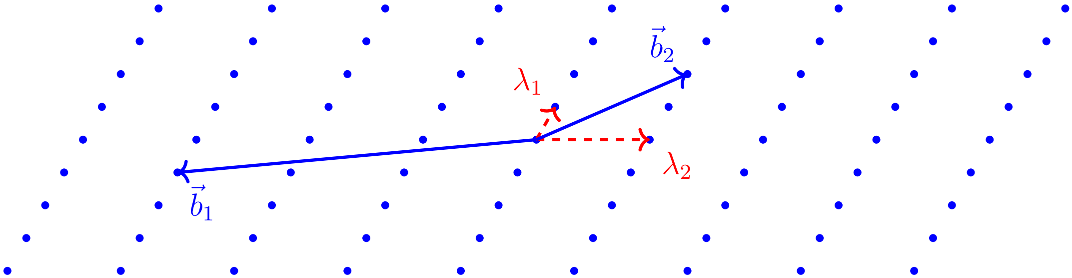


Vecteurs Courts



- Un réseau étant discret, sa longueur minimale est bien définie. On la note λ_1.
- De même, on définit λ_i comme la longueur du plus court vecteur linéairement indépendant des vecteurs atteignant les $i-1$ premiers minima.
- **Théorème (Minkowski) :** $\lambda_1(L) \leqslant \sqrt{n} \cdot \det L^{1/n}$

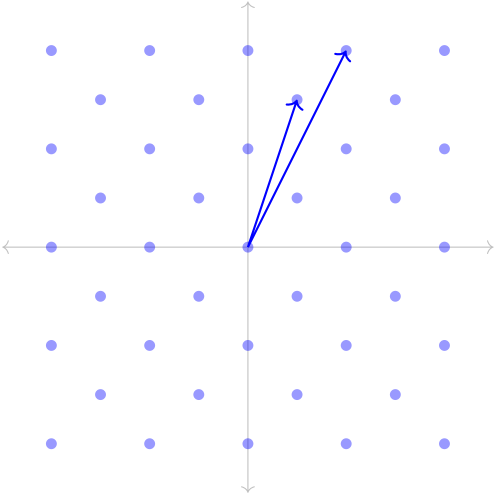
Trouver un Vecteur Court



- Déterminer un vecteur atteignant λ_1 est très difficile, même en pratique.
- On a alors recourt à des algorithmes dits de « Réduction de réseau ».

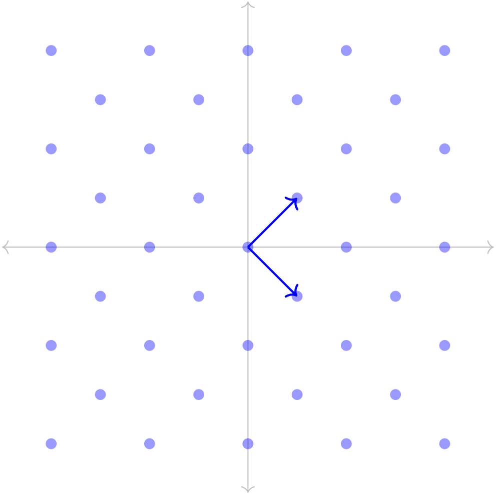
Réduction de Réseau

Base Aléatoire



$$\xrightarrow{LLL}$$

Base Réduite



Algorithme LLL

Lenstra, Lenstra, Lovász (LLL) :
Étant donnée une base d'un réseau $\mathcal{L}$, il est possible de trouver en *temps polynomial* une nouvelle base de $\mathcal{L}$ telle que

- $||b_i|| \leqslant 2^{(n-1)/2}\lambda_i$
- $||b_1|| \leqslant 2^{(n-1)/4} \det \mathcal{L}^{1/n}$

(Nguyen, Stehlé) : En pratique, LLL parvient à trouver des vecteurs

$$||v|| \approx (1.02)^n \cdot \det \mathcal{L}^{1/n}.$$

LLL est difficile à implémenter correctement (problème de précisions). La meilleure approche est appelée *flatter* **(Ryan, Heninger 2023)** et est implantée dans Pari/GP par exemple.

Applications à la Cryptanalyse

Another Lattice Attack Against an RSA-like Cryptosystem

George Teşeleanu[1,2]

RSA et Polynômes

Rappel :

- La clé publique est (N, e) avec $N = p \cdot q$ et $\mathrm{PGCD}(e, (p-1) \cdot (q-1)) = 1$.
- Un message M est chiffré par $C \equiv M^e \mod N$.

M est racine de $x^e \; - \; C$ dans $\mathbb{Z}/N\mathbb{Z}$.

Comment Trouver des Racines de Polynômes ?

Trouver les racines entières d'un polynôme de $\mathbb{Z}[x]$?

Trouver les racines rationnelles d'un polynôme de $\mathbb{Q}[x]$?

Trouver les racines d'un polynôme de $\mathbb{Z}/p\mathbb{Z}[x]$ (p premier) ?

Comment Trouver des Racines de Polynômes ?

Trouver les racines entières d'un polynôme de $\mathbb{Z}[x]$?

$\rightarrow$ Approximation numérique, et arrondi à l'entier le plus proche.

Trouver les racines rationnelles d'un polynôme de $\mathbb{Q}[x]$?

Trouver les racines d'un polynôme de $\mathbb{Z}/p\mathbb{Z}[x]$ (p premier) ?

Comment Trouver des Racines de Polynômes ?

Trouver les racines entières d'un polynôme de $\mathbb{Z}[x]$?
$\rightarrow$ Approximation numérique, et arrondi à l'entier le plus proche.

Trouver les racines rationnelles d'un polynôme de $\mathbb{Q}[x]$?
$\rightarrow$ Algorithme LLL (c'était la motivation première)

Trouver les racines d'un polynôme de $\mathbb{Z}/p\mathbb{Z}[x]$ (p premier) ?

Comment Trouver des Racines de Polynômes ?

Trouver les racines entières d'un polynôme de $\mathbb{Z}[x]$?
$\rightarrow$ Approximation numérique, et arrondi à l'entier le plus proche.

Trouver les racines rationnelles d'un polynôme de $\mathbb{Q}[x]$?
$\rightarrow$ Algorithme LLL (c'était la motivation première)

Trouver les racines d'un polynôme de $\mathbb{Z}/p\mathbb{Z}[x]$ (p premier) ?
$\rightarrow$ Facile, *e.g.*, algorithmes de Berlekamp, ou Cantor-Zassenhaus.

Racines modulo N

Si $N = p_1 \cdots p_\ell$ et la factorisation est connue, il suffit de trouver les racines modulo chacun des p_i.

Si $N = p_1^{e_1} \cdots p_\ell^{e_\ell}$ et que les e_i ne sont pas trop gros, on peut se ramener au cas précédent et utiliser ce qu'on appelle le « relevé de Hensel » (Hensel lifting).
Attention : Le nombre de racines peut-être exponentiel! (exemple $x^d \mod p^d$).

Cas général : Se ramène essentiellement à factoriser N.

Mise en Bouche : Petite Racine

On cherche des racines de $F(x) \stackrel{\text{def}}{=} x^e - C$ dans $\mathbb{Z}/N\mathbb{Z}$

Remarque : Si $|M| < N^{1/e}$, alors $C \equiv M^e \mod N = M^e$ dans $\mathbb{Z}$!

Mise en Bouche : Petite Racine

On cherche des racines de $F(x) \stackrel{\text{def}}{=} x^e - C$ dans $\mathbb{Z}/N\mathbb{Z}$

Remarque : Si $|M| < N^{1/e}$, alors $C \equiv M^e \mod N = M^e$ dans $\mathbb{Z}$!

Généralisation : Si $G(x) = x^d + a_{d-1}x^{d-1} + \cdots + a_1 x + a_0 \in \mathbb{Z}[x]$ est tel qu'il existe x_0 avec

- $G(x_0) \equiv 0 \mod M$;
- $|x_0| < M^{1/d}$;
- Les a_i sont « suffisamment petits » ;

Alors $G(x_0) = 0$ sur $\mathbb{Z}$.

Mise en Bouche : Petite Racine

On cherche des racines de $F(x) \overset{\text{def}}{=} x^e - C$ dans $\mathbb{Z}/N\mathbb{Z}$

Remarque : Si $|M| < N^{1/e}$, alors $C \equiv M^e \mod N = M^e$ dans $\mathbb{Z}$!

Généralisation : Si $G(x) = x^d + a_{d-1}x^{d-1} + \cdots + a_1 x + a_0 \in \mathbb{Z}[x]$ est tel qu'il existe x_0 avec

- $G(x_0) \equiv 0 \mod M$;
- $|x_0| < M^{1/d}$;
- Les a_i sont « suffisamment petits » ;

Alors $G(x_0) = 0$ sur $\mathbb{Z}$.

Idée : Trouver un G relié à F vérifiant ces propriétés.

Un Exemple

$$\boxed{M = 17 \cdot 19 = 323 \text{ et } F(x) = x^2 + 33x + 215}$$

On définit $G(x) \stackrel{\text{def}}{=} 9F(x) - M(x+6) = 9x^2 - 26x - 3$.

- Si $F(x_0) = 0 \mod M$ alors $G(x_0) = 0 \mod M$.
- On cherche x_0 dans les racines entières de G.
- On vérifie que $x_0 = 3$ convient.

Généralisation : l'idée de Coppersmith

On suppose que $0 \leqslant M < N$ et que l'on connaît une proportion $1 - \frac{1}{e}$ bits de poids forts de M : on écrit $M = \mu + M_0$ où μ est connu et $|M_0| < N^{1/e}$.

Idée : Alors $C \equiv (\mu + M_0)^e \mod N$
i.e., M_0 est une petite racine de $(\mu + x) - C$ dans $\mathbb{Z}/N\mathbb{Z}$.

Le Théorème de Coppersmith

Coppersmith (1996) :
Soit N un entier à la factorisation inconnue, et soit $f \in \mathbb{Z}[x]$ unitaire, de degré d.
Alors il est possible de retrouver **tous** les entiers x_0 qui vérifient

- $f(x_0) \equiv 0 \mod N$
- $|x_0| < N^{1/d}$.

Le Théorème de Coppersmith

Coppersmith (1996) :
Soit N un entier à la factorisation inconnue, et soit $f \in \mathbb{Z}[x]$ unitaire, de degré d.
Alors il est possible de retrouver **tous** les entiers x_0 qui vérifient

- $f(x_0) \equiv 0 \mod N$
- $|x_0| < N^{1/d}$.

Quel lien avec les réseaux Euclidiens ?

Un Exemple

Supposons que

$$F(x) = x^3 + f_2x^2 + f_1x + f_0 \in \mathbb{Z}[x].$$

On cherche $G(x) \in \mathbb{Z}[x]$ avec **petits coefficients** ayant les mêmes racines modulo N.

Idée : Chercher $G(x) = A(x) \cdot F(x) + B(x) \cdot N \in \langle F(x), N \rangle$

Un Exemple

Supposons que

$$F(x) = x^3 + f_2x^2 + f_1x + f_0 \in \mathbb{Z}[x].$$

On cherche $G(x) \in \mathbb{Z}[x]$ avec **petits coefficients** ayant les mêmes racines modulo N.

Si on se limite au degré 3, on écrit

$$\begin{aligned} G(x) &= c_3f(x) + N \cdot (c_2x^2 + c_1x + c_0) \\ &= c_3x^3 + (c_3f_2 + c_2N)x^2 + (c_3f_1 + c_1N)x + c_3f_0 + c_0N. \end{aligned}$$

Un Exemple

Si on se limite au degré 3, on écrit

$$G(x) \quad = c_3x^3 + (c_3f_2 + c_2N)x^2 + (c_3f_1 + c_1N)x + c_3f_0 + c_0N.$$

Le vecteur des coefficients de G est alors de la forme

$$\begin{pmatrix} c_0 & c_1 & c_2 & c_3 \end{pmatrix} \cdot \begin{pmatrix} N & & & \\ & N & & \\ & & N & \\ f_0 & f_1 & f_2 & 1 \end{pmatrix}$$

On cherche un **vecteur court** dans un réseau!

La Méthode de Coppersmith

Entrée : $F(x) \in \mathbb{Z}[x]$ et $N \in \mathbb{Z}$.
Sortie désirée : x_0 tel que $F(x_0) \equiv 0 \mod N$
Étape intermédiaire : $G(x) \in \mathbb{Z}$ tel que $G(x_0) = 0$ sur $\mathbb{Z}$.

(1) Construire une matrice représentant le réseau des polynômes de degré borné dans $\langle F(x), N \rangle$.
(2) Appliquer un algorithme de réduction de réseaux.
(3) Construire un polynôme G à partir du plus petit vecteur obtenu.
(4) Récupérer les racines de G.

$\boxed{\text{Comment s'assurer que } G \text{ a ses racines dans } \mathbb{Z}\,?}$

Lemme d'Howgrave-Graham

Howgrave, Graham (1997) :

Soit $F(x) \stackrel{\text{def}}{=} \sum_{i=0}^{d} a_i x^i \in \mathbb{Z}$ et soit $N, R \in \mathbb{Z}$. On suppose que $x_0 \in \mathbb{Z}$ vérifie $|x_0| \leqslant R$ et $F(x_0) \equiv 0 \mod N$.

On note $b_F \stackrel{\text{def}}{=} (a_0, a_1 R, a_2 R^2, \ldots, a_d R^d)$.

Si $\|b_F\|_2 < \dfrac{N}{\sqrt{d+1}}$ alors $F(x_0) = 0$ dans $\mathbb{Z}$.

Remarque sur la norme :

$$\begin{aligned} |F(x_0)| &= |a_0 + a_1 x_0 + a_2 x_0^2 + \cdots + a_d x_0^d| \\ &\leqslant |a_0| + |a_1| R + |a_2| R^2 + \cdots + |a_d| R^d = \|b_F\|_1. \end{aligned}$$

Construire le Réseau

On définit le réseau $\mathcal{L}_F$ de base

$$B \stackrel{\text{def}}{=} \begin{pmatrix} N & 0 & \cdots & 0 & 0 \\ 0 & NR & \cdots & 0 & 0 \\ \vdots & & & \vdots & \vdots \\ 0 & 0 & \cdots & NR^{d-1} & 0 \\ a_0 & a_1 R & \cdots & a_{d-1}R^{d-1} & R^d \end{pmatrix}$$

Il est de rang $d+1$ et de déterminant $\det \mathcal{L}_F = |\det B| = ?$

Construire le Réseau

On définit le réseau $\mathcal{L}_{F,N}$ de base

$$B \stackrel{\text{def}}{=} \begin{pmatrix} N & 0 & \cdots & 0 & 0 \\ 0 & NR & \cdots & 0 & 0 \\ \vdots & & & \vdots & \vdots \\ 0 & 0 & \cdots & NR^{d-1} & 0 \\ a_0 & a_1 R & \cdots & a_{d-1}R^{d-1} & R^d \end{pmatrix}$$

Il est de rang $d+1$ et de déterminant

$$\det \mathcal{L}_{F,N} = |\det B| = N^d R^{d(d+1)/2}.$$

Proposition de Correction (Faible)

Soit $G(x)$ le polynôme dont les coefficients forment le premier vecteur de la base LLL réduite de $\mathcal{L}_{N,F}$. Si

$$R < \frac{(d+1)^{-1/d} \cdot N^{2/d(d+1)}}{\sqrt{2}}$$

alors toute racine x_0 de $F(x)$ modulo N telle que $|x_0| \leqslant R$ vérifie $G(x_0) = 0$ dans $\mathbb{Z}$.

Preuve : $\|G\|_2 \leqslant 2^{((d+1)-1)/4} \det(L)^{1/(d+1)} (*) \leqslant 2^{d/4} N^{d/(d+1)} R^{d/2}$ et on applique Howgrave-Graham.

- Si $d = 3$ (RSA) il suffit que $R \approx N^{1/6}$.
- Pour atteindre $N^{1/d}$ il faut un peu plus travailler.
- En pratique LLL marche mieux que cette borne $(*)$.

Coppersmith pour RSA

Entrée : $F(x) = (x+\mu)^3 - C = x^3 + 3\mu x^2 + 3\mu^2 x + \mu^3 - C$ et $N \in \mathbb{Z}$.
Sortie désirée : $|x_0| < R$ tel que $F(x_0) \equiv 0 \mod N$

- On construit le réseau $\begin{pmatrix} N & 0 & 0 & 0 \\ 0 & NR & 0 & 0 \\ 0 & 0 & NR^2 & 0 \\ \mu^3 - C & 3\mu^2 R & 3\mu R^2 & R^3 \end{pmatrix}$
- $\dim \mathcal{L} = 4$ et $\det \mathcal{L} = R^6 N^3$.

Conclusion

Si on connaît une proportion $1 - \frac{1}{e}$ des bits du messages, alors on peut le retrouver.

$\Rightarrow$ Retrouver une proportion $1 - \frac{1}{e}$ des bits du message est aussi dur que de retrouver tout le message !

La cryptanalyse à base de réseaux dépasse RSA et Coppersmith :

- Cryptanalyse de problèmes de type « Sac-à-dos » (*Knapsack*).
- Cryptanalyse de la fonction de hachage de Damgård (Joux, 1994).
- Problème du « nombre caché » (courbes elliptiques).
- ...