Cryptanalyse de crypto-système par réduction de réseaux

Aloïs Colléaux-Le Chêne

24/04/2023

LRE - EPITA

Plan

1. Travaux précédent
2. Définitions des termes
3. Attaque existante sur ECDH
4. Améliorations
5. Conclusion
6. Questions ouvertes

Travaux précédents sur ECDSA[1]

Algorithm 1 Message signature in ECDSA

Input: (E, G, n), a point G of order n on the elliptic curve E

Input: $(d_A, Q_A = [d_A]G)$, the private-public keypair

Input: m, the message to sign

Output: (r, s), the signature

1: $e \leftarrow \text{hash}(m)$ ▷ hash is a cryptographic hash function

2: **repeat**

3: $k \xleftarrow{\$} [1, n-1]$ ▷ k is also called the nonce

4: $(x_1, y_1) \leftarrow [k]G$

5: $r \leftarrow x_1 \mod n$

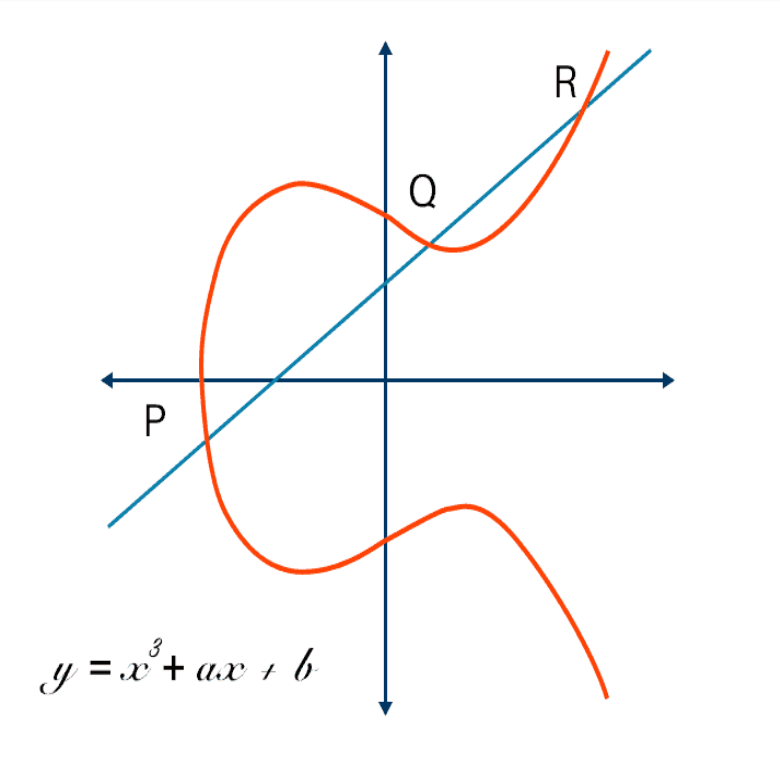
6: **until** $r \neq 0$

7: $s \leftarrow k^{-1}(e + rd_A) \mod n$

Travaux précédents sur ECDSA[1]

- Le nonce doit rester secret
- Fuite du nonce = récupération de la clé
- Fuite partielle = récupération de la clé, après calculs
- Limite dure sur les algorithmes de SVP et CVP
- Le nonce se retrouve à partir du vecteur le plus court d'un réseau
- Introduction du problème $\mathrm{BDD}_{\alpha,\mathrm{f}(\cdot)}$

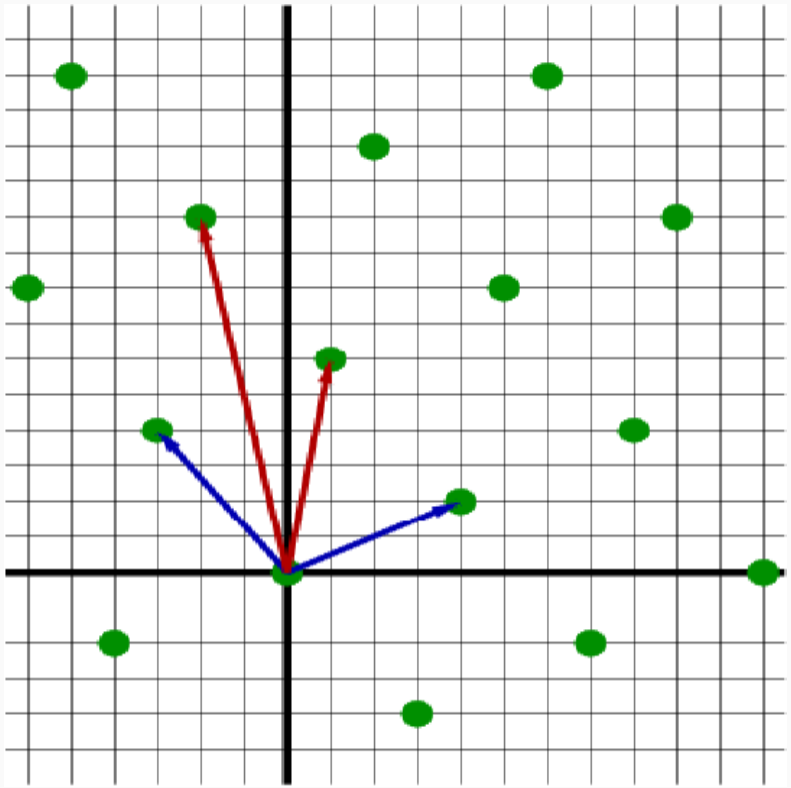
Courbe elliptiques



- Une courbe planaire sur un corps fini
- Ensemble des points satisfaisant $y^2 = x^3 + ax + b$
- Différentes opérations sur des points

Réseaux euclidiens

- Ensemble des combinaisons linéaires de vecteurs d'une base dans $\mathbb{Z}^n$
- On note $\Lambda = \{\sum_{i=1}^{n} a_i v_i \mid a_i \in \mathbb{Z}\}$, où $\{v_1, \ldots, v_n\} \subset \mathbb{Z}^n$ est une base pour Λ, le réseau.



Réseaux euclidiens: notations

- On note $\mathbf{B}$ la matrice carrée ayant pour lignes la base du réseau
- Inversement, on note $\Lambda(\mathbf{B})$ le réseau engendré par les lignes de $\mathbf{B}$
- $\lambda_n(\Lambda)$ correspond au rayon de la boule centrée à l'origine contenant n vecteurs non-colinéaires

Heuristique de Gauss

- Estimation du nombre de points d'un réseau Λ dans $S \subset \mathbb{R}^n$
- $|S \cap \Lambda| \approx \mathrm{vol}(S)/\det(\mathbf{B})$
- Peut être utilisé pour estimer la taille du vecteur le plus court
- $\lambda_1(\Lambda) \approx \mathrm{GH}(\Lambda) = \frac{(\Gamma(n/2+1)(\det \Lambda))^{1/n}}{\sqrt{\pi}}$

Problèmes commun de réseaux

- Problème du vecteur le plus court (SVP)
  - Soit Λ un réseau, trouver le vecteur le plus court de Λ
  - C'est-à-dire trouver un vecteur de norme $\lambda_1(\Lambda)$
  - LLL, BKZ, HKZ
- Problème du vecteur le plus proche (CVP)
  - Soit Λ un réseau, $\mathbf{x} \in \mathbb{R}^n$, trouver $\mathbf{y} \in \Lambda$ tel que $\mathbf{y} = \min \operatorname{dist}(\mathbf{x}, \mathbf{v}), \mathbf{v} \in \Lambda$
  - Peut être plongé en instance SVP
  - Babai's rounding algorithm

α-Bounded Distance Decoding (BDD$_\alpha$)

- Déclinaison du CVP
- Avec une garantie: $\text{dist}(\mathbf{x}, \Lambda) < \alpha \cdot \lambda_1(\Lambda(B))$
- Garantie d'unicité si $\alpha < 1/2$
- Probabilité d'unicité élevée si $1/2 \leq \alpha < 1$

α-Bounded Distance Decoding with predicate ($\mathrm{BDD}_{\alpha,\mathsf{f}(\cdot)}$)

- À partir d'un vecteur $\mathbf{x} \in \mathbb{R}^n$ et d'un prédicat $f(\cdot)$, trouver le vecteur $\mathbf{y} \in \Lambda$ le plus proche, tel que $f(\mathbf{y} - \mathbf{x}) = \top$
- La garantie de distance à Λ reste la même
- Garantie d'unicité grâce au prédicat
- Ne donne plus forcément le vecteur le plus proche

unique Shortest Vector Problem with Predicate ($\mathrm{uSVP}_{f(\cdot)}$)

- Déclinaison du SVP
- Soit un prédicat $f(\cdot)$, trouver le vecteur le plus court $\mathbf{x} \in \Delta$ tel que $f(\mathbf{x}) = \top$
- Garantie d'unicité grâce au prédicat
- Ne donne pas forcément le vecteur le plus court

Elliptic-Curve Diffie-Hellman (ECDH)

- Un protocole d'échange de clés, sur courbe elliptique
- Paramètres du domaine: (p, a, b, G, n)
  - Les opérations se passent sur E, la courbe $y^2 = x^3 + ax + b \mod p$
  - G est générateur d'un sous-groupe de E, et son cardinal est n
- La clé privée est un entier aléatoire d et la clé publique $Q = [d]G$
- On note la paire de clés d'Alice (d_a, Q_a) et celle de Bob (d_b, Q_b)
- Analogue au Diffie-Hellman classique, le point secret partagé est $S = d_a \cdot Q_b = d_b \cdot Q_a$.
- On utilise la composante x du secret.
  - La composante y ne donne qu'un seul bit d'information

Problème du nombre caché (HNP)

- Soit un entier secret α et un module publique n, des informations à propos de α sont révélées ainsi
  - Un oracle choisis un entier uniformément aléatoire, $0 < t_i < n$, calcule $s_i = t_i \cdot \alpha \mod n$ et renvoie des bits de poids forts de s_i et t_i
  - On note $s_i = a_i + k_i$, où $k < 2^l$
  - l est un paramètre de difficulté du problème
- On cherche à trouver α
- Calculer les bits de poids forts de la clé est aussi difficile que de calculer la clé dans l'échange de clé Diffie-Hellman
- Solvable avec des algorithmes de SVP avec la technique du plongement de Kannan

Problème du nombre caché, sur courbe elliptique (EC-HNP$_x$)

Definition

EC-HNP$_x$[2] Soit un premier p, une courbe elliptique E sur Fp, un point $R \in E$ et un entier positif δ. On pose $P \in E$ un point secret. Soit $O_{P,R}$ un oracle, tel que à partir m renvois les δ bits de poids fort de la composante x de $P + [m]R$. C'est-à-dire, $O_{P,R}(m) = MSB_\delta(x_{P+[m]R})$.

L'objectif est de retrouver le point secret P, grâce à un accès à l'oracle.

Casser ECDH

- **Récupérer un secret partagé**
- Fuite partielle lors des l'opération
- Clé de la forme $a + m$
- On observe plusieurs échanges de secret
- On récupère les bits de poids fort du secret partagé
  - $[(a + m)b]Q = [ab]Q + [m][b]Q = P + [m]R$

Partie en ligne

On récupère n échantillons

On pose $Q = [m]R$

$$
\begin{aligned}
h_0 &= \mathrm{MSB}_k(P_x) = P_x - e_0 \\
h_i &= \mathrm{MSB}_k((P+Q)_x) = (P+Q)_x - e_i \\
h'_i &= \mathrm{MSB}_k((P-Q)_x) = (P-Q)_x - e'_i \\
\tilde{h}_i &= h_i + h'_i \\
\tilde{e}_i &= e_i + e'_i
\end{aligned}
$$

Le polynôme bivarié

$$F_i(X, Y) := X^2 Y + A_i X^2 + A_{0,i} XY + B_i X + B_{0,i} Y + C_i$$

$$F_i(e_0, \tilde{e}_i) \equiv 0 \mod p$$

Le polynôme bivarié: le retour

$$
\begin{aligned}
A_i &= (\tilde{h} - 2Q_x) \\
A_{0,i} &= 2(h_0 - Q_x) \\
B_i &= 2[\tilde{h}(h_0 - Q_x) - 2h_0 Q_x - a - Q_x^2] \\
B_{0,i} &= (h_0 - Q_x)^2 \\
C_i &= [\tilde{h}(h_0 - Q_x)^2 - 2h_0^2 Q_x - 2(a + Q_x^2)h_0 - 2aQ_x 4b]
\end{aligned}
$$

Plongement vers $\mathrm{uSVP}_{\mathrm{f}(\cdot)}$

$$\mathbf{M} = \begin{pmatrix} \boldsymbol{\Delta}^2 & 0 & M_1 \\ 0 & \boldsymbol{\Delta} & M_2 \\ 0 & 0 & P \end{pmatrix}$$

Plongement vers $\mathrm{uSVP}_{f(\cdot)}$: les détails

$$M_1 = \begin{pmatrix} -C_1 & -C_2 & \dots & -C_n \\ -B_1 & -B_2 & & -B_n \\ -B0,1 & 0 & & 0 \\ 0 & -B_{0,2} & & \\ \vdots & & \ddots & \\ & 0 & & \\ & \vdots & & \\ 0 & 0 & & -B_{0,n} \end{pmatrix}, M_2 = \begin{pmatrix} -A_1 & -A_2 & & -A_n \\ -A0,1 & 0 & & 0 \\ 0 & -A_{0,2} & & \\ \vdots & & \ddots & \\ & 0 & & \\ & \vdots & & \\ 0 & 0 & & -A_{0,n} \end{pmatrix}$$

Plongement vers $\mathrm{uSVP}_{f(\cdot)}$: un exemple

$$M = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -C_1 & -C_2 \\
0 & 2^{k-m} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -B_{0,1} & 0 \\
0 & 0 & 2^{k-m} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -B_{0,2} \\
0 & 0 & 0 & 2^{k-m} & 0 & 0 & 0 & 0 & 0 & -B_1 & -B_2 \\
0 & 0 & 0 & 0 & 2^{2(k-m)} & 0 & 0 & 0 & 0 & -A_{0,1} & 0 \\
0 & 0 & 0 & 0 & 0 & 2^{2(k-m)} & 0 & 0 & 0 & 0 & -A_{0,2} \\
0 & 0 & 0 & 0 & 0 & 0 & 2^{2(k-m)} & 0 & 0 & -A_1 & -A_2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 2^{3(k-m)} & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2^{3(k-m)} & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & p & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & p
\end{pmatrix}$$

Plongement vers $\mathrm{uSVP}_{f(\cdot)}$: un exemple

$$\mathbf{v} := \langle 1, \tilde{e_1}, \ldots, \tilde{e_n}, e_0\tilde{e_1}, \ldots, e_0\tilde{e_n}, e_0^2, e_0^2\tilde{e_1}, \ldots, e_0^2\tilde{e_n}, k_1, \ldots, k_n \rangle$$

- $\det(\mathbf{M}) = \frac{p^n}{2^{(m-k)(6n+3)}}$
- $|\mathbf{vM}| \leq \sqrt{3n+3} \ll \mathrm{GH}(\Lambda(\mathbf{M}))$
- $\mathrm{GH}(\Lambda(\mathbf{M})) = \sqrt{4n+3}(2^{(k-m)(6n+3)p^n})^{\frac{1}{4n+3}}$

$\mathbf{vM}$ est le vecteur le plus court $\iff 2^k \gg 2^{5/6m}$

Breaking the ECHNP lattice barrier

Utilisation de $\text{uSVP}_{\text{f}(\cdot)}$ au lieu SVP

Prédicat: $\text{f} : (v_0, \ldots, v_{2n+3}) \rightarrow v_0 \neq 0$

L'algorithme

Algorithm 2 Find P_x

Input: $O_{P,R}$, the EC-HNP$_x$ oracle

Output: x_P

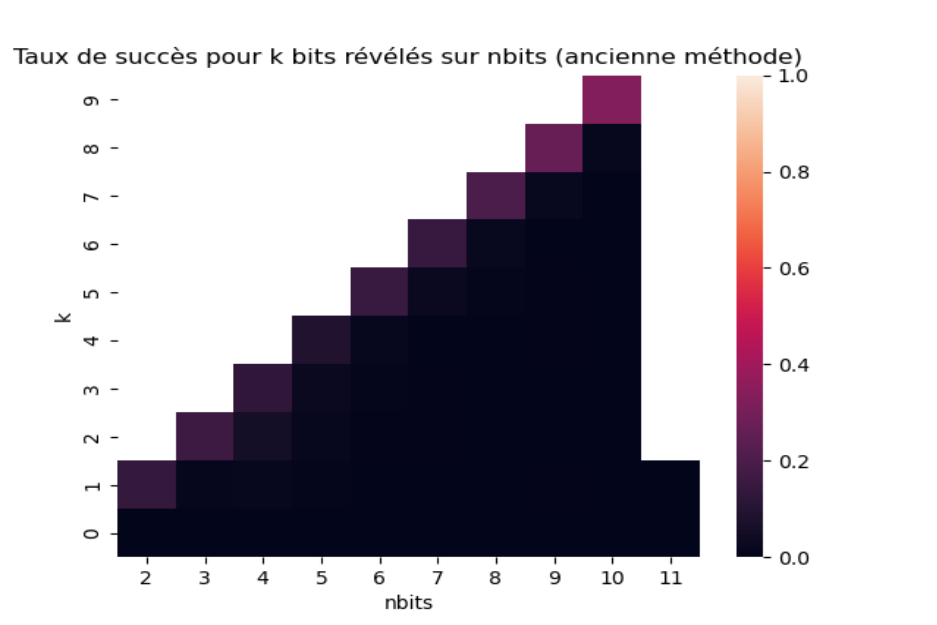
Construct the matrix $\mathbf{M}$

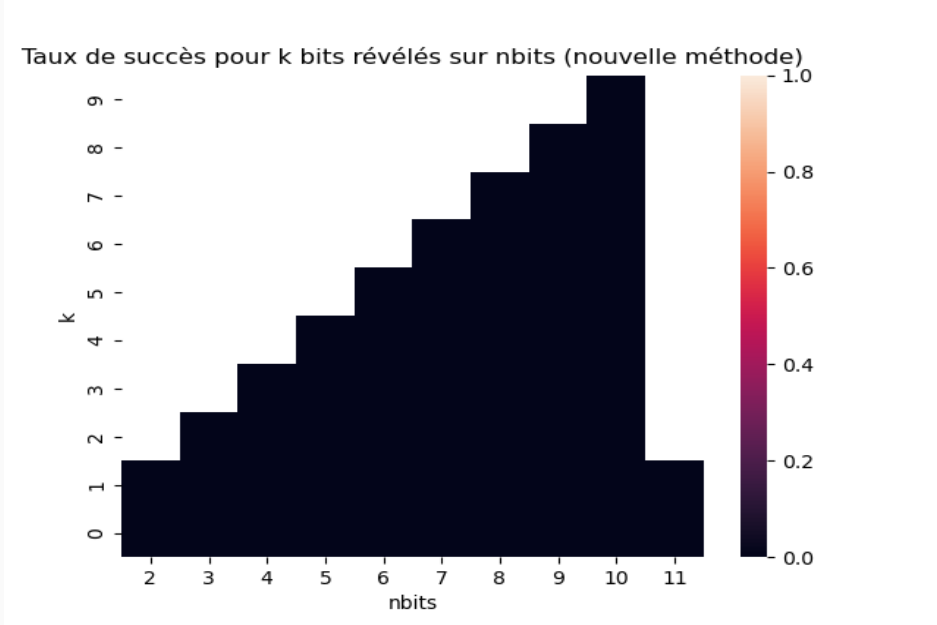
Let f : $(v_0, \ldots, v_{2n+3}) \rightarrow v_0 \neq 0$, the predicate

Let $\boldsymbol{f} \leftarrow \text{uSVP}_{\text{f}(\cdot)}(\Lambda(\mathbf{M}))$

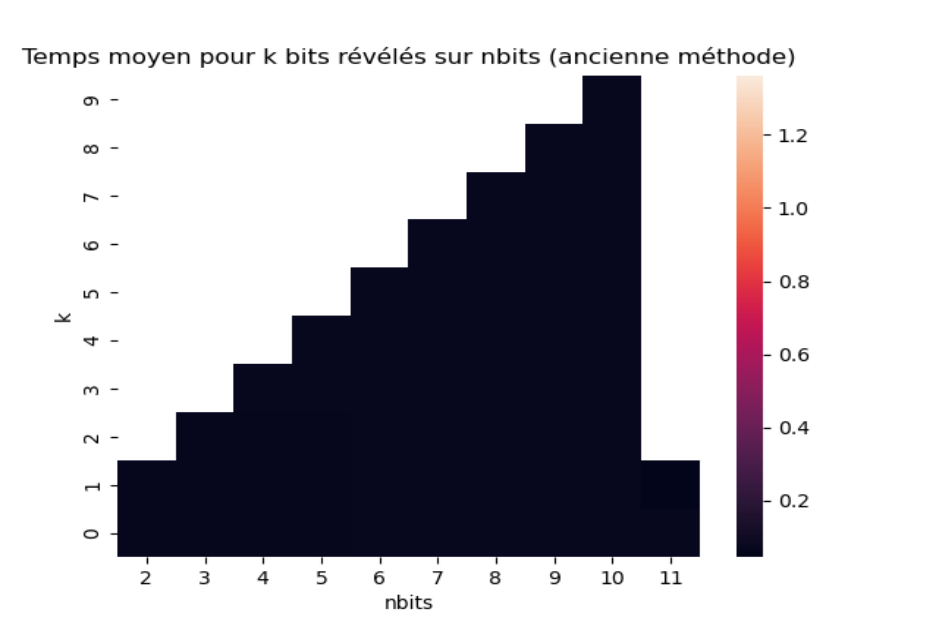
return $h_0 + \frac{f_0}{f_0'}$

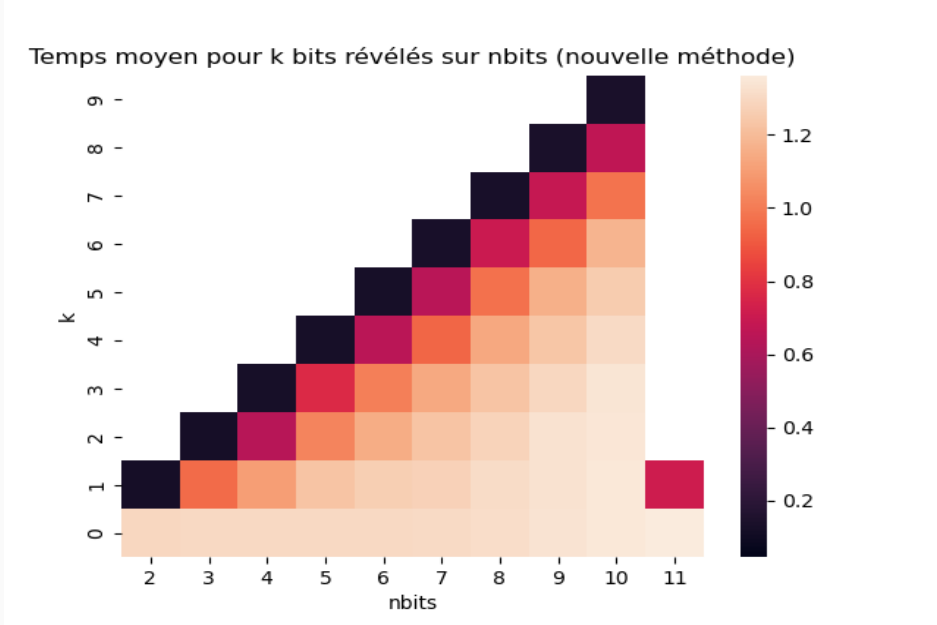
Résultats


Taux de succès pour k bits révélés sur nbits (ancienne méthode)
k
nbits
0
1
2
3
4
5
6
7
8
9
2
3
4
5
6
7
8
9
10
11
1.0
0.8
0.6
0.4
0.2
0.0

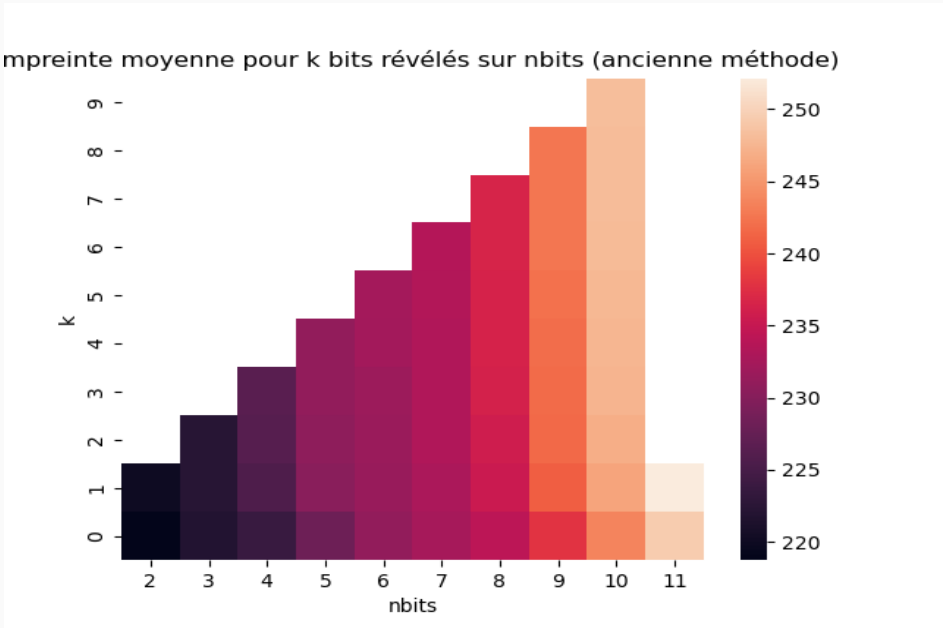

Taux de succès pour k bits révélés sur nbits (nouvelle méthode)
k
nbits
0
1
2
3
4
5
6
7
8
9
2
3
4
5
6
7
8
9
10
11
1.0
0.8
0.6
0.4
0.2
0.0

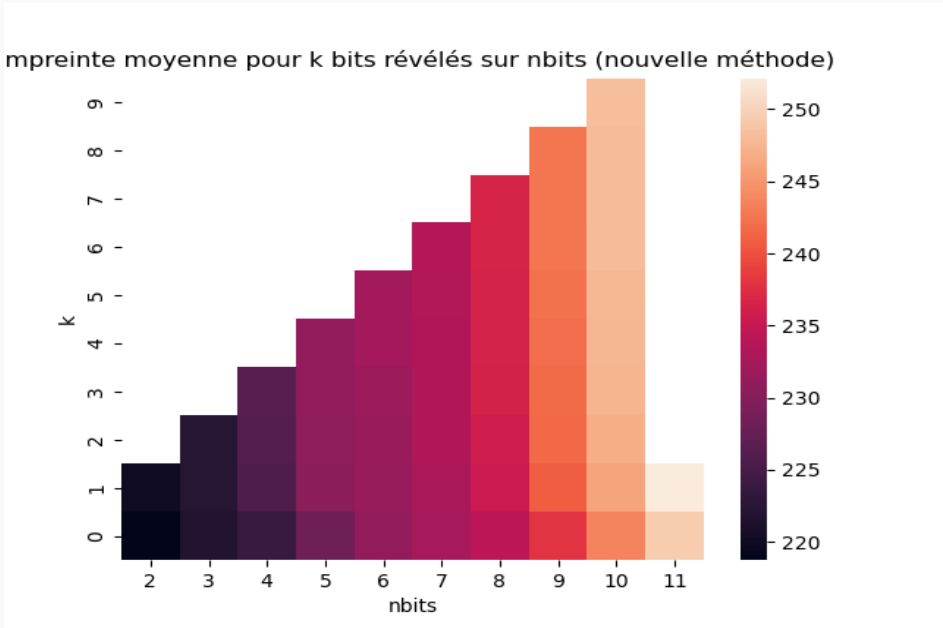
Performances





Consommation mémoire





Conclusion

- Un meilleur taux de succès
- Un prix trop élevé pour être applicable
- De bons espoirs pour son amélioration

Questions ouvertes

1. Trouver la valeur optimale de n (i.e. la dimension du réseau)
2. Changer la structure de la base[3]
3. Changer le prédicat
4. Choisir les m des échantillons

Bibliographie


Martin R. Albrecht and Nadia Heninger.
On bounded distance decoding with predicate: Breaking the "lattice barrier" for the hidden number problem.
Cryptology ePrint Archive, Paper 2020/1540, 2020.
https://eprint.iacr.org/2020/1540.


Barak Shani.
On the bit security of elliptic curve diffie–hellman.
Cryptology ePrint Archive, Paper 2016/1189, 2016.
https://eprint.iacr.org/2016/1189.


Jun Xu, Santanu Sarkar, Huaxiong Wang, and Lei Hu.
Improving bounds on elliptic curve hidden number problem for ecdh key exchange.
Cryptology ePrint Archive, Paper 2022/1239, 2022.
https://eprint.iacr.org/2022/1239.

Clap de fin

Merci pour votre attention !