Лабораторная работа № 3

Факторизация на решётках

Дедлайн: 03.04.2023

1 Алгоритм факторизации Шнорра

Лабораторная работа вдохновлена недавнем пре-принтом К.П. Шнорра [1], вызвавший большой резонанс [2, 3, 4]. В частности, аннотация статьи утверждает, что "This [атака] destroys the RSA cryptosystem." Суть этой лабораторной - попытаться факторизовать RSA модуль с помощью решёток.

Алгоритм имеет длинную историю [5], мы будем придерживаться описания из [6]

В основе алгоритма (как и для продвинутых алгоритмов Number Field Sieive), лежит идея поиска пар (x, y), удовлетворяющих

$$x^2 \equiv y^2 \bmod N. \tag{1}$$

Если $x \neq \pm y \bmod N$, то вычисление $\gcd(N, x + y)$ даст нетривиальный делитель N. Метод Шнорра ищет такие пары с доп. условие гладкости.

Число называется B-*гладким*, если все его простые делители меньше B. Обозначим p_i–i-ое простое число и зафиксируем некоторое целое $d > 1$. Основная вычислительная задача алгоритма Шнорра состоит в поиске четверок (u, v, k, γ), таких что 1. u, v, k–p_d-гладкие; 2. $\gamma \geq 1$ – целое; 3. выполняется Диофантово уравнение

$$u = v + kN^\gamma.$$

Эти четверки будут находится с помощью коротких векторов решетки специального вида. А именно, рассмотри решетку, порожденную столбцами матрицы A:

$$A = \begin{pmatrix} \ln p_1 & 0 & \dots & 0 & 0 \\ 0 & \ln p_2 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \ln p_d & 0 \\ C \ln p_1 & C \ln p_2 & \dots & C \ln p_d & C \ln N \end{pmatrix} \in \mathbb{Z}^{d+1 \times d+1},$$

где C–некая (достаточно большая) константа. Для вектора $\mathbf{z} \in \mathbb{Z}^{d+1}$, справедливо

$$A\mathbf{z} = \begin{pmatrix} z_1 \ln p_1 \\ \vdots \\ z_d \ln p_d \\ C(\sum_i z_i \ln p_i + z_{d+1} \ln N). \end{pmatrix}$$

Если мы будем ассоциировать с вектором $\mathbf{z}$, элементы u, k, γ следующим образом

$$u = \prod_{z_i > 0} p_i^{z_i}, \quad k = \prod_{z_i < 0} p_i^{-z_i} \quad \text{и} \quad \gamma = |z_{d+1}|,$$

то

$$\|A\mathbf{z}\|_1 = \sum_i^d |z_i| \ln p_i + C \left| \sum_i^d z_i \ln p_1 - |z_{d+1}| \ln N \right|,$$

и

$$\|A\mathbf{z}\|_1 = \ln u + \ln k + C\,|\ln u - \ln(kN^\gamma)|\,.$$

Отсюда, если $\|A\mathbf{z}\|_1$ – мала, то можно доказать следующее утверждение (см. [6] для доказательства):

Если $\|A\mathbf{z}\|_1 \le 2C + 2\sigma \ln p_d - \gamma \ln N$, то $|u - kN^\gamma| < p_d^\sigma$.

Делаем вывод: чтобы найти четверку (u, v, k, γ), необходимо найти короткий вектор в решетке, порожденной столбцами A. Как найти из такой четверки (x, y), удовлетворяющие 1?

Найдем $d+1$ четверок (u, v, k, γ). Для каждой такой четверки положим $a_{i,j} = z_j$ для $z_j > 0$ (те z_j, что участвуют в записи u_i), а за $b_{i,j}$ обозначим степени в разложении $v_i = u_i - k_i N_i^\gamma = \prod_i p_i^{b_{i,j}}$ (для $i \le d+1$). Обозначим далее вектора $\mathbf{a}_i = (a_{i,1}, \ldots, a_{i,d})$, $\mathbf{b}_i = (b_{i,1}, \ldots, b_{i,d})$. Всего имеем $(d+1)$ таких векторов, которые будем записывать в матрицу $M \in \mathbb{Z}^{d \times d+1}$ по столбцам.

Для всякого вектора $\mathbf{c} \in \{0,1\}^{d+1}$, удовлетворяющего

$$\mathbf{c} \cdot M \equiv 0 \bmod 2,$$

положим

$$x = \prod_{j=1}^{d} p_j^{\sum_{i=1}^{d+1} c_i (a_{i,j} + b_{i,j})/2} \bmod N \qquad y = \prod_{j=1}^{d} p_j^{\sum_{i=1}^{d+1} c_i a_{i,j}} \bmod N.$$

Если $x \neq \pm y \bmod N$, вычислим нетривиальный делитель N как $\gcd(N, x+y)$.

1.1 Предложение Шнорра 2021

Алгоритм, описанный выше, требует поиска нескольких (минимум $d+1$) коротких векторов для решетки, порожденной матрицей A. Для этого, например, можно использовать алгоритм просеивания, возвращающий (почти) все короткие вектора решетки [7] (см examples/all_short_vectors.py).

Шнорр в [1] предлагает рандомизировать A следующим образом. Выбираем случайную перестановку $f : [1, d] \to [1, d]$ и строим A'

$$A' = \begin{pmatrix} Nf(1) & 0 & \ldots & 0 & 0 \\ 0 & Nf(2) & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & 0 \\ 0 & 0 & \vdots & Nf(n) & 0 \\ CN\ln p_1 & CN \ln p_2 & \ldots & CN \ln p_d & CN \ln N, \end{pmatrix}$$

для какого-то (большого) C. Найдя короткий вектора этой решетки $A\mathbf{z}$, "уберем" скаляр C, положив $\mathbf{z}' = \mathbf{z}/C$. Представим

$$u = \prod_{i:\mathbf{z}'_i > 0} p_i^{\mathbf{z}'_i} \qquad k = \prod_{i:\mathbf{z}'_i < 0} p_i^{-\mathbf{z}'_i}.$$

Из найденных значений (u, v) строим (x, y) аналогично процедуре выше, *если* $(u - kN)$ — p_d−гладкое число.

1.2 Задание

Используя **любой** из подходов и их улучшений (смотрим ссылки), факторизовать 30-битный RSA-модуль N (то есть $N = pq$, где $p \neq q$ – простые числа по ≈15 бит каждое).

Бонус. Команде[1], успешно факторизовавшей 80-битный RSA-модуль N алгоритмом Шнорра, положен +1 балл на экзамене.

[1] Команда состоит из 1-2 человек

Список литературы

[1] Claus Peter Schnor. *Fast Factoring Integers by SVP Algorithms.* https://eprint.iacr.org/2021/232.pdf

[2] https://twitter.com/inf_0_/status/1367376526300172288

[3] https://twitter.com/kennyog/status/1367132559117848583

[4] https://crypto.stackexchange.com/questions/88582/does-schnorrs-2021-factoring-method-show-that-t
88647#88647

[5] https://github.com/lducas/SchnorrGate

[6] Antonio Vera. *A note on integer factorization using lattices* https://arxiv.org/pdf/1003.5461.pdf

[7] https://github.com/fplll/g6k