lattice_crypto.sage

SC4010 ‘Presentation’{

[Lattice-based Cryptography]

< Wong Chu Feng
Wan Kai Jie >

}

Sage

lattice_crypto.sage

Contents

Inspired by:

Twenty Years of Attacks on the RSA Cryptosystem

— Dan Boneh (1999)

01

[Introduction]

RSA & Lattice-based attacks

RSA

Strength from hard factoring problem of N=p*q

Lattice-based attacks

- application of lattice structures
- find smallest vectors by basis reduction
- attacks weak RSA when e is small

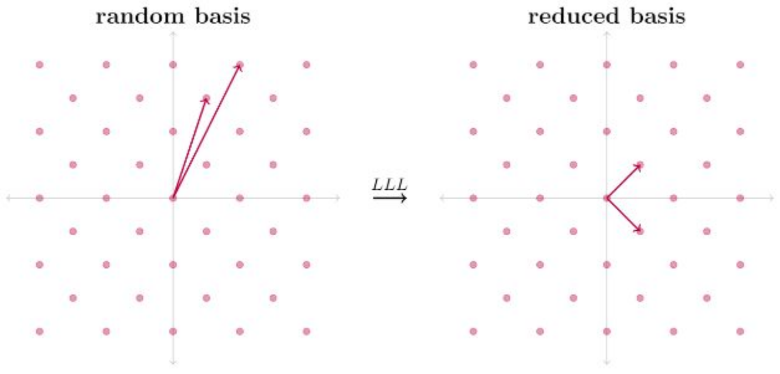
02

[Coppersmith Attack]

LLL, Howgrave-Graham

Lenstra-Lenstra-Lovász (LLL)

- Approximation to the Shortest Vector Problem (SVP)
- SVP (hard problem) → shortest nonzero vector
- Reduces basis vectors (of lattice) into shorter and more orthogonal vectors using Gram-Schmidt process



Gram-Schmidt Process

- Reduces basis vectors (of lattice) into shorter and more orthogonal vectors using Gram-Schmidt process

$$\text{Given basis: } \mathcal{B} = \{u_1, \ldots, u_k\}$$

- Set $v_1 \triangleq u_1$
- For $j = 2, \ldots, k,$

$$v_j \triangleq u_j - \frac{\langle u_j, v_1 \rangle}{\langle v_1, v_1 \rangle} v_1 - \ldots - \frac{\langle u_j, v_{j-1} \rangle}{\langle v_{j-1}, v_{j-1} \rangle} v_{j-1}$$

- Normalise all v_i.

$$\text{Set } \mathcal{A} \triangleq \left\{ \frac{v_1}{\|v_1\|}, \ldots, \frac{v_k}{\|v_k\|} \right\}$$

Coppersmith theorem

- Find small solutions to polynomial equations
- Powerful when parts of the Ciphertext is known to attacker

Theorem Let N be an integer and $f \in \mathbb{Z}[x]$ be a monomic polynomial of degree d. Set $X = N^{\frac{1}{d}-\epsilon}$ for some $\epsilon \geq 0$. Then, given $\langle N, f \rangle$, Eve can efficiently find all integers $|x_0| < X$ satisfying $f(x_0) = 0 \pmod{N}$. The running time is dominated by the time it takes to run the LLL algorithm on a lattice of dimension $O(w)$ with $w = min\{\frac{1}{\epsilon}, log_2 N\}$.

Coppersmith theorem

- Our simplification

Assume prime p: $p = a + r$, a is known, where $N = p * q$

1. Define polynomial $f(x) = \sum_{i=0}^{n-1} a_i x^i + x^n$
2. Build matrix using coefficients from f
3. LLL algorithm reduces basis vectors to construct a new polynomial g s.t. $g(r) = 0$ on the integers
4. Then test each potential root: $r_i | N$

Howgrave-Graham

- States that when scaled by X, Euclidean norm must be less than an upper bound
- X upper bound limits the size of the root we are trying to find
- That is root x_0 can be found if g(x_0) is sufficiently small

Theorem Let $g(x)$ be an univariate polynomial with n monomials. Further, let $m \in \mathbb{Z}^+$. Suppose that

- $g(x_0) = 0 \pmod{N^m}$ where $|x_0| \leq X$
- $\|g(xX)\| < \frac{N^m}{\sqrt{n}}$

Then $g(x_0) = 0$ holds over the integers.

Condition:
$2^{w/4} det(L)^{\frac{1}{w}} < \frac{N^m}{\sqrt{w}}$, where w is $dim(Matrix)$

Howgrave-Graham

- Construct matrix with coefficients from polynomial
- Then perform LLL algorithm and extract the new polynomials g row by row.
- Test if each polynomial g(x), the root x_0 exists to satisfy f(x_0) = 0 (mod N^m)

Let $f(x) = a_0 + a_1 x + a_2 x^2 + x^3$, then matrix M will be

$$M = \begin{bmatrix} a_0 & a_1 X & a_2 X^2 & X^3 \\ 0 & a_0 X & a_1 X^2 & a_2 X^3 \\ 0 & 0 & a_0 X^2 & a_1 X^3 \\ 0 & 0 & 0 & a_0 X^3 \end{bmatrix}$$

03

[Demo]

[References]

- Small Solutions to Polynomial Equations, and Low Exponent RSA Vulnerabilities (Don Coppersmith 1995)
- https://www.youtube.com/watch?v=3cicTG3zeVQ
- https://github.com/mimoo/RSA-and-LLL-attacks.git
- Mathematics of Public Key Cryptography Chapter 19 (Steven D Galbraith 2018)
- Coppersmith/Howgrave-Graham and LLL (Tanja Lange 2020)

[Contributions]

Wan Kai Jie:

Wong Chu Feng:

[Thank you!]