Introduction to Cryptology

13.3 - The Index Calculus

Federico Pintore
Mathematical Institute, University of Oxford (UK)

UNIVERSITY OF OXFORD

Michaelmas term 2020

L notation

$$L_Q(\alpha; c) = \exp((c + o(1))(\log Q)^{\alpha}(\log\log Q)^{1-\alpha})$$

- $\alpha = 0 \Rightarrow L_Q(\alpha; c) = (\log Q)^{(c+o(1))}$ (polynomial in $\| Q \|$).
- $\alpha = 1 \Rightarrow L_Q(\alpha; c) = Q^{(c+o(1))}$ (exponential $\| Q \|$).

Index Calculus for $\mathbb{Z}_p^*$

Problem: given $g, h \in \mathbb{Z}_p^*$, find x such that $h = g^x$.

- Fix some bound $B \in \mathbb{N}$, and let $\mathcal{F} = \{p_1, \ldots, p_k\}$ be the set of primes less than or equal to B.
- Relation search
  - Compute $g_i := g^{a_i}$ for random $a_i \in \{1, \ldots, p-1\}$.
  - If g_i is B-smooth, then

$$g^{a_i} \pmod{p} = \prod_{j=1}^{k} p_j^{e_{i,j}}. \tag{1}$$

- Linear algebra Once $\ell \geq k$ linearly independent equations of the form (1) are found, solve for $\log_g p_i$, $i = 1, \ldots, k$, modulo $p-1$.
- Search for t such that $g^t \cdot h \pmod{p}$ is B-smooth. Once found, solve for $\log_g h \mod (p-1)$.

Complexity Analysis

We assume that the cost of generating relations dominates the overall complexity of the algorithm.

If B is large, it is more likely that the g_i are B-smooth, but more relations are necessary. The two costs need to be balanced.

The prime number theorem says that

$$k = |\{\text{primes } p_i \leq B\}| \approx \frac{B}{\log_e B}.$$

Complexity Analysis

- Define $\Psi(N, B) = |\{B\text{-smooth positive integers} \leq N\}|$, with $N = p - 1$.

- The probability that a positive integer $m \leq N$ is B-smooth is approximately equal to $\frac{1}{N} \cdot \Psi(N, B)$.

- <u>Canfield-Erdos-Pomerance Theorem</u>:

  Let $u = \frac{\log_e N}{\log_e B}$. Then $\frac{1}{N} \cdot \Psi(N, N^{1/u}) = u^{-u+o(u)} \approx u^{-u}$.

- The expected number of random exponentiations necessary to find a B-smooth g_i is $\approx u^u$.

Complexity Analysis

The expected running time of the algorithm is

$$\approx \underbrace{(k+1)}_{\text{nb of relations}} \cdot \underbrace{u^u}_{\text{expected nb of trials}} \cdot \underbrace{k}_{\text{nb of trial divisions}} \cdot \underbrace{M(\log_e N)}_{\text{time for a trial division}}$$

$$\approx B^2 \cdot u^u \qquad \left(\text{drop the logarithmic factors, where } k \approx \frac{B}{\log_e B}\right)$$

$$= N^{2/u} \cdot u^u \qquad (u = \log_e N / \log_e B \Rightarrow N = B^u)$$

Goal: minimize $f(u) = N^{2/u} \cdot u^u$

Complexity Analysis

An *approximate* minimum is reached for u s.t. $u^2 \log u \approx 2 \log_e N$.

For $u = 2\sqrt{\dfrac{\log_e N}{\log_e \log_e N}}$, it holds:

$$u^2 \log_e u = 4\frac{\log_e N}{\log_e \log_e N}\left(\log_e 2 + \frac{1}{2}\log_e \log_e N - \frac{1}{2}\log_e \log_e \log_e N\right)$$

and therefore $u^2 \log_e u = 2 \log_e N + o(\log_e N)$.

Complexity Analysis

The value of u makes B equal to:

$$\begin{aligned} B &= N^{1/u} \\ &= \exp\left(\frac{1}{u}\log_e N\right) \\ &= \exp\left(\frac{1}{2}\sqrt{\log_e N \log_e \log_e N}\right) \\ &= L_N(1/2, 1/2) \end{aligned}$$

Note that $u^u = L_N(1/2, 1)$.

Therefore $B^2 u^u = L_N(1/2, 2)$.

The cost of the linear algebra step is bounded by $\tilde{O}(B^3)$ (which is $O(B^3 \log_e N)$), i.e. $L_N(1/2, 3/2)$.

Further Reading I

Andrew Granville.
Smooth numbers: computational number theory and beyond.
Algorithmic number theory: lattices, number fields, curves and cryptography, 44:267–323, 2008.

Antoine Joux, Andrew Odlyzko, and Cécile Pierrot.
The past, evolving present, and future of the discrete logarithm.
In Open Problems in Mathematics and Computational Science, pages 5–36. Springer, 2014.

Carl Pomerance.
Smooth numbers and the quadratic sieve.
Algorithmic Number Theory, Cambridge, MSRI publication, 44:69–82, 2008.

Further Reading II

Carl Pomerance.
A tale of two sieves.
Biscuits of Number Theory, 85, 2008.

Victor Shoup.
Lower bounds for discrete logarithms and related problems.
In Advances in Cryptology—EUROCRYPT'97, pages 256–266. Springer, 1997.