Ruhr University Bochum
Chair for Cryptology and IT-Security
Prof. Dr. Alexander May
Andre Esser

RUB

Kaliningrad Summerschool 2019

Day II Exercises on DLP, Collision Finding and Subset Sum

Exercise 1:

Given k discrete logarithm instances in the same group: $\beta_i := g^{x_i}$, $i = 1, \ldots, k$, show how to compute all x_i in time $\tilde{\mathcal{O}}(\sqrt{k \cdot |G|})$

Exercise 2:

Given a discrete logarithm instance $(g, \beta = g^x)$. State an algorithm computing x in time $\tilde{\mathcal{O}}(x^{\frac{3}{2}})$ and memory $\tilde{\mathcal{O}}(1)$.

Exercise 3:

Consider Floyd's cycle finding algorithm for finding a collision in a random function $f\colon \{0,1\}^n \to \{0,1\}^n$ in time $\tilde{\mathcal{O}}(2^{\frac{n}{2}})$. Show how this algorithm can be used to find a collision between two random n-bit functions f_1, f_2 using expected time $\tilde{\mathcal{O}}(2^{\frac{n}{2}})$.

Exercise 4:

In the following we investigate the subset sum problem, which is defined as follows: For a given random vector $\mathbf{a} = (a_1, a_2, \ldots, a_n)$ with $a_i \in_R \mathbb{Z}_{2^n}$ and $t \in \mathbb{Z}_{2^n}$ the goal is to find a vector $\mathbf{e} \in \{0,1\}^n$ of weight $\mathrm{wt}(\mathbf{e}) = \alpha n$, $\alpha \in \left[0, \frac{1}{2}\right]$ satisfying $\langle \mathbf{a}, \mathbf{e} \rangle = t \bmod 2^n$ for a given α. In other words we are looking for a subset of the a_i of size αn that sums up to $t \bmod 2^n$.

1) Give a Meet-in-the-Middle algorithm on $\mathbf{e}$ with time and memory complexity $\tilde{\mathcal{O}}(2^{\frac{H(\alpha)n}{2}})$.
2) Devise an algorithm based on collision finding with time complexity $\tilde{\mathcal{O}}(2^{\frac{3H(\alpha)n}{4}})$ and memory complexity $\tilde{\mathcal{O}}(1)$.
3) Show how to find the solution in time $\tilde{\mathcal{O}}\left(2^{\left(\frac{3}{2}H(\alpha/2)-\alpha\right)n}\right)$ using only polynomial memory.