Cryptography

JOACHIM VON ZUR GATHEN, JÉRÉMIE DETREY

4. Assignment: Computing discrete logarithms

(Hand in solutions on Tuesday, December 11th during the lecture)

Exercise 4.1 (Baby-step giant-step). (17 points)

The objective of this exercise is to re-discover and understand a classical algorithm for computing discrete logarithms: the *baby-step giant-step* algorithm.

Before tackling the general algorithm, let's look at a simple example: we consider the multiplicative group $\mathbb{Z}_{25}^{\times}$ and the primitive element $g = 2$.

(i) How many elements are there in $\mathbb{Z}_{25}^{\times}$? $\boxed{1}$

(ii) Using trial multiplication, compute $\text{dlog}_2(6)$, the discrete logarithm of 6 in base 2. $\boxed{1}$

We now want to compute the discrete logarithm of $x = 19$. But instead of using the trial multiplication method, we will use the baby-step giant-step method.

For this purpose, we first fix an integer m. For instance, let's take $m = 5$ here.

(iii) Compute the value of $x \cdot g^i$, for i from 0 to $m - 1$. These are called *baby steps*. $\boxed{2}$

(iv) Now, compute the value of g^{jm} for j from 0 to $\lceil (\#\mathbb{Z}_{25}^{\times})/m \rceil - 1$. These are the *giant steps*. $\boxed{2}$

(v) Find a collision between baby steps and giant steps. Namely, a pair of integers i and j such that $x \cdot g^i = g^{jm}$. $\boxed{1}$

(vi) Conclude on the discrete logarithm of 19 in base 2. $\boxed{2}$

Now on to the general case.

(vii) Describe an algorithm for computing discrete logarithms based on the baby-step giant-step approach shown above. Keep m as a parameter of this algorithm. $\boxed{4}$

(viii) What is the complexity (in terms of group operations and storage requirements) of this algorithm (depending on m)? 2

2 (ix) What is the best choice for m?