Cryptography, winter term 16/17: Assignment 9

Prof. Markus Bläser, Cornelius Brand, Marc Roth
http://www-cc.cs.uni-saarland.de/course/55/

Due January 18th, 2017

Exercise 9.1 (Be nice to your tutors and TAs, 1 Bonus Point) Write the name and matriculation number of every author as well as number, time slot and the name of the tutor of your tutorial group on the first page of your solution. Furthermore write in english and staple *all* sheets of your solution.

Exercise 9.2 (RSA and Factoring, 2+1+2+3 Points) The following exercises can and should be solved without using an electronic calculator.

(a) Calculate $\gcd(352, 17)$ via the extended euclidian algorithm and compute s and t such that $\gcd(352, 17) = s \cdot 352 + t \cdot 17$.

(b) Compute $d := 17^{-1} \bmod \Phi(391)$. **Hint:** 23 is a divisor of 391.

(c) Compute x such that $x^{17} = 49 \bmod 391$. You may use that $49^{29} = 2 \bmod 391$.

(d) Prove that hardness of *RSA-Inv* implies hardness of *Factoring*, i.e., assume the existence of a ppt adversary $\mathcal{A}$ that wins the Factoring-Experiment with non-negligible probability and use it to construct a ppt adversary that wins the RSA-Inv experiment with non-negligible probability.

Exercise 9.3 (Miller-Rabin-Test, 4 Points + 2 Bonus Points) The proof of correctness of the Miller-Rabin primality test you saw in the lecture left two gaps open, which we will address in this exercise. As usual, inputs will be coded in binary, so to represent a natural number t you need at most $\log t + 1$ bits. This is important: You will have to show that your algorithms run in polynomial time in the *size* of the numbers given as input (and not in the absolute value of the input numbers, which might be exponentially larger than the number of bits needed to represent it). You may assume without proof that arithmetic operations on numbers can be performed in polynomial time in their size. Keep in mind that you still have to argue that the resulting numbers stay of polynomial size in the size of n (e.g., 2^{2^n} is not computable in polynomial time).

(a) Show how to decide whether a given natural number $n > 1$ is a perfect power[1] in polynomial time.
Hint: Try to bound the maximal value of the exponent b polynomially in the size of n. You can then go through all exponents iteratively, and in each step decide in polynomial time in the size of n whether there is a basis a for the current exponent b, using a suitable search algorithm.

[1] Recall that a natural number $n \in \mathbb{N}$ is called a *perfect power* if there are $a, b \in \mathbb{N}$ such that $a^b = n$ and $a, b > 1$.

(b) **Bonus:** Given a natural number $n \in \mathbb{N}$ and $a \in \mathbb{Z}_n^*$, show how to decompose $n-1 = 2^r u$ with u odd and $r \geq 1$ in polynomial time (argue why $r \geq 1$ when executing the Miller-Rabin test!) Then, show how to decide whether a is a strong witness that n is composite in polynomial time.
Hint: Review the exercises from last week's presence exercise sheet. Do you see one that seems related?

Do not forget to prove correctness and any claimed bound on the running time of your algorithms!

Exercise 9.4 (RSA and Hash functions, 1+2+1 Points) Consider the following hash function (Gen, H):

- Gen: On input 1^n we run $\text{GenRSA}(1^n)$ to get N, e and d. We choose $y \in \mathbb{Z}_N^* \setminus \{1\}$ uniformly at random and output $s = (N, e, y)$.
- H: Given a key $s = (N, e, y)$ and an input $x \in \{0,1\}^{3n}$, H^s is defined as

  $$H^s(x) := f^s_{x_{3n}}(f^s_{x_{3n-1}}(\dots f^s_{x_1}(1)\dots))$$

  where $f_0^s, f_1^s : \mathbb{Z}_N^* \to \mathbb{Z}_N^*$ such that

  $$f_0^s(a) := [a^e \bmod N] \text{ and } f_1^s(a) := [a^e \cdot y \bmod N]\,.$$

  Note that the image of f_1^s, f_0^s is $\mathbb{Z}_N^*$ as this group is closed under multiplication (modulo N).

In this exercise you are asked to prove that hardness of *RSA-Inv* (relative to GenRSA) implies that (Gen, H) is collision-resistant. It may be advisable to proceed as follows

a) Show that f_0^s and f_1^s are bijections for fixed s.

b) Now assume a collision $a \neq b$ was found, i.e. $H^s(a) = H^s(b)$. Define

   $$a^0 = 1 \text{ and } a^{i+1} = f^s_{a_{i+1}}(a^i)$$

   and

   $$b^0 = 1 \text{ and } b^{i+1} = f^s_{b_{i+1}}(b^i)\,.$$

   Prove that there is an index j such that (i) $a^j \neq b^j$, (ii) $a^{j+1} = b^{j+1}$, and (iii) $a_{j+1} \neq b_{j+1}$.

c) Conclude that there are distinct u and v such that $f_0^s(u) = f_1^s(v)$ and show how to construct x from u and v such that

   $$x^e = y \bmod N\,.$$

Example: Assume $s = (1111, 0111, 0100)$, i.e. $N = 15, e = 7$ and $y = 4$. Furthermore, let $x = 011100110001$. Then

$$H^s(x) = f_1^s(f_0^s(f_0^s(f_0^s(f_1^s(f_1^s(f_0^s(f_0^s(f_1^s(f_1^s(f_1^s(f_0^s(1)))))))))))$$

where

$$f_0^s(a) := [a^7 \bmod 15] \text{ and } f_1^s(a) := [a^7 \cdot 4 \bmod 15]\,.$$