Heads and tails, summer 2015

PROF. DR. JOACHIM VON ZUR GATHEN AND DR. DANIEL LOEBENBERGER

1. Exercise sheet
Hand in solutions until Saturday, 18 April 2015, 23:59:59

For the course we remind you of the following dates:

- Lectures: Monday 13:00h-14:30h and Thursday 12:45-14:15h, B-IT 2.1.
- Tutorial: Monday 14:45h-16:15h, B-IT 2.1.

Exercise 1.1 (NIST generator with backdoor). (14+7 points)

A simplified variant of the *dual-elliptic-curve pseudorandom generator*, standardized in the NIST Special Publication 800-90A: "Recommendation for Random Number Generation Using Deterministic Random Bit Generators", works as follows: Fix an n-bit prime p. We are given an elliptic curve E over $\mathbb{F}_p$ of (prime) order d together with two points $P, Q \in E$ on it. For a finite point $P = (x, y)$ let $\mathrm{x}(P) = x$ be the x-coordinate of P. We start from a secret random integer seed s_0. For each $i \geq 1$ the generator calculates $s_i = \mathrm{x}(s_{i-1}P)$ and $t_i = \mathrm{x}(s_iQ)$, outputs t_i and starts anew with seed s_i.

(i) Draw a schematic picture of the generator. 2

Suppose you know $a = \mathrm{dlog}_P Q$, so that $Q = aP$.

(ii) Prove that $\mathrm{x}(a^{-1}s_iQ) = \mathrm{x}(s_iP) = s_{i+1}$, where the inversion is done modulo d. Hint: On an elliptic curve the x-coordinates of P and $-P$ are equal. 4

(iii) Explain how to compute $\mathrm{x}(a^{-1}s_iQ)$. 2

(iv) Argue that you have broken the generator. 2

On the webpage you find the above mentioned NIST Special Publication 800-90A.

(v) Look into the standard and explain the major differences between the above description and the standardized generator. 4

(vi) Break the standardized generator in a similar way as above. Argue about the efficiency of your attack. How can you prevent it? +7

Exercise 1.2 (Hardware generator). (17+5 points)

In the course we discussed the physical random generator `PRG310-4`. On our course webpage you find two 500KB sample outputs: The first file contains raw output only, the second a second output which is AES-postprocessed.

7 (i) Compute for both files the distributions of bytes, i.e. count for every byte how often it occurs in the file. Hand in a graphical representation of the resulting distribution.

2 (ii) What is the maximal possible entropy for a distribution on 256 items?

5 (iii) Which entropy values do you obtain for the two distributions?

3 (iv) What do you observe?

+5 (v) What is the purpose of cryptographic postprocessing in general? Which requirements have to be fulfilled? You might want to consult the BSI document AIS31 given on the course webpage.