Cryptography

Priv.-Doz. Dr. Adrian Spalka, Konstantin Ziegler

5 Assignment

(Due: Thursday, 02 December 2010, 12^{00})

Exercise 5.1 (block ciphers)**.** (10 points) A block cipher encrypts messages of a fixed block length, e.g. 128 bits in the case of AES. If shorter or longer messages have to be handled, we have to apply

- *padding* to extend the message length to a multiple of the block length and
- *modes of operation* to specify how the sequence of blocks is processed.

(i) List four modes of operation and their advantages and disadvantages.

(ii) Suggest a mode of operation which can *not* be found on wikipedia and advertise it.

(iii) Which key length do we have to choose for RSA and EC-crypto in order to achieve security similar to AES-128. Use implementations of them on your machine to time the encryption of 1KB, 10KB, 100KB, ...

Exercise 5.2 (stream ciphers)**.** (10 points) A stream cipher may encrypt a stream of message bits by XORing with a stream of cipher bits derived from the secret key. If the message is as long as the secret key, we can choose the key as cipher stream and obtain the *one-time-pad*. However, in many applications, the length of the message exceeds the length of the secret key and/or is unknown when encryption starts.

Given a key string $k_0k_1 \dots k_{n-1}$ of n bits and a linear polynomial $f(x_0, x_1, \dots, x_n)$ in $n+1$ variables over $\mathbb{Z}_2$, where the coefficient of x_n is nonzero, we define a stream s_i of bits by

$$s_i = \begin{cases} k_i & \text{if } i < n, \\ f(s_{i-n}, s_{i-n+1}, \dots, s_{i-1}, 0) & \text{else.} \end{cases}$$

(i) For a moment, think of integers instead of bits and find an initial key and a polynomial that generate the Fibonacci sequence.

A sequence s_i as described above is a *linearly recurrent sequence*. The smallest integer L such that $s_i = s_{i+L}$ for all i is called *least period*.

(i) Determine L for (lots of) different n and f and describe the pattern you find. (Hint: View f as a polynomial in only one variable: $f(x^0, x^1, \ldots, x^n)$.)

(ii) How many bits of output do you have to intercept in order to recover the secret key?

(iii) Such linearly recurrent sequence can be easily generated by *linear feedback shift registers*. They are fast, but cryptographically weak as seen above. How does a *Geffe generator* help?

Exercise 5.3 (mathematical bonus)**.** (+2 points) How many quadratic residues are in $\mathbb{Z}_N$?