Security

Assignment 3, Friday, September 23, 2016

Handing in your answers: For the full story, see

`http://www.sos.cs.ru.nl/applications/courses/security2016/exercises.html`

To summarize:

- Include your name and student number **in** the document (they will be printed!), as well as the name of your teaching assistant (Hans or Joost). When working together, include **both** your names and student numbers.
- Submit one single **pdf** file – when working together, only hand in **once**.
- Hand in via Blackboard, before the deadline.

Deadline: Monday, October 3, 09:00 sharp!

Goals: After completing these exercises successfully you should be able to

- use the Vigenère cipher and one-time pad;
- encrypt and decrypt using an LFSR;
- understand the relation between the Vigenère cipher and one-time pad.

Marks: You can score a total of 100 points.

1. **(30 points)** During the lecture, the Vigenère was introduced. For more background information, see e.g. `http://en.wikipedia.org/wiki/Vigenere_cipher`.

   (a) Decrypt the following Vigenère ciphertext using the key '`franklin`'.

   ```
   yyrroxilpveckdmpwvtvpeebtwtuoxiejuenn
   ```

   (b) Consider the following modification to the Vigenère cipher: Instead of specifying a single word as key, one gives a starting point in a book, such as "Open the book '1984' on page 4; the key starts at the 13rd word on this page". The key consists of continuous text starting from the specified word and is *as long as the message to encrypt.*
   Encrypt the following text with this method (page 4 is available[1] in the preview on Amazon's site `http://www.amazon.co.uk/Nineteen-Eighty-Four-George-Orwell/dp/0141393041/` – click the 'Excerpt' or 'Look inside' link, the key thus starts with "there was no way. . . "):

   ```
   War is peace.
   Freedom is slavery.
   Ignorance is strength.
   ```

   Remove all spaces and punctuation in both the key and the plaintext and convert all upper-case characters to lower case.

   (c) Even though this is stronger than Vigenère with a repeating key, why is this still not secure (for sufficiently long messages)? Sketch the idea of an attack against this cipher!

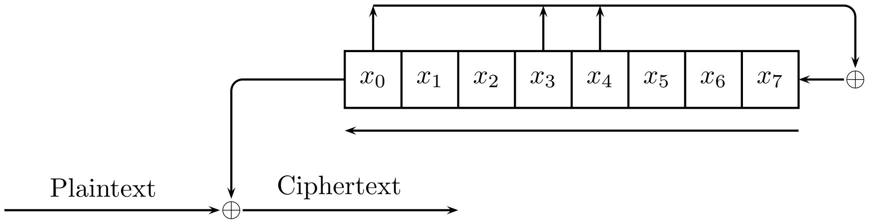
[1] The Penguin 2008 mass-market paperback edition was used for this exercise.

2. **(40 points)** The one-time pad scheme is a very secure encryption scheme, but it has one important disadvantage: The pad can only be used *once*.

   In this exercise, you get a ciphertext that resulted from a one-time pad encryption, as well as some parts of the plaintext and the key stream. Additionally, there is a weakness you can exploit: The key stream used to create this ciphertext was not used only one time, but a part of it has been used multiple times. Using this knowledge, recover the plaintext. Note that the repetition can start at any point in the pad, and the bits at the start of the pad are not necessarily part of the repeating pattern. Once the repetition has started, it continues forever. In order to be able to perform an XOR operation on the bits, the characters in the plaintext are translated to 7-bit ASCII binary representation[2] (e.g. 'a' becomes `1100001`).

ASCII	`u`	`s`	`e`		...	`u`	`s`	...	...	`o`	`n`	...	...	...	...	...	...
plain	1110101	1110011	...	0100000	1101010	...	...	...	0100000	1101111	...	...	...	...	...	...	...
pad	1010011	0011000	...	...	...	...	1001011	0110010	...	0001000	...	...	...	...	...	...	...
XOR	0100110	1101011	1011000	...	1001010	1110100	...	1000110	...	...	1101110	...	...	...	0101011	1101101	...
ASCII	`&`	`k`	`X`	`8`	`J`	`t`	`8`	`F`	`*`	`g`	`n`	`W`	`L`	`6`	`+`	`m`	`i`

3. **(30 points)** Consider the following simple Linear Feedback Shift Register (LFSR). The plaintext is bitwise XOR-ed with the output bits of the LFSR which **first computes** $x_0 \oplus x_3 \oplus x_4$ and **then shifts** such that x_0 falls out.



   Example:

   The initial state **0** 1 1 **1** **1** 0 1 1

   is followed by **1** 1 1 **1** **0** 1 1 0

   and outputs **0**

   (a) Describe the next five states of the LFSR if it is initialized according to the box above. The first successor state is already given as illustration, so you have to give the four subsequent ones.

   (b) Also do a "rollback" and compute the *previous* four states, starting from the initial state.

   (c) Assume you know that the LFSR is in the initial state given above. After four shifts you intercept 0100 as resulting ciphertext. Reconstruct the 4 bits of plaintext that were encrypted to this ciphertext.

[2] `http://en.wikipedia.org/wiki/ASCII`