Cryptography, homework sheet 6

Due: 15 October 2015, 10:45

Team up in groups of two or three to hand in your homework. We do not have capacity to correct all homeworks individually. To submit your homework, email the programming part to `crypto15@tue.nl` and place the written part on the lecturer's table before the lecture. Do not email Tanja or put homework in mailboxes.

You may use computer algebra systems such as mathematica, gp, or sage or program in C, Java, or Python. Please submit your code if any) as part of your homework. If you do, make sure that your programs compile and run correctly; my students will not debug your programs. The program should also be humanly readable.

Last call: If you are a student in the old masters system and for some reason cannot substitute 2MMC10 for the course (2WC09 or 2WC12) in your study program, please contact Tanja by email. To find out whether you can substitute the course, talk to your study advisor.

1. A message of length 64 bytes is encrypted with AES and sent via a network. During the transmission one bit in the second block is flipped. Explain for each of the 5 modes of operation

   (a) how many bits are potentially different in the deciphered text compared to the initial plaintext;

   (b) how many bits are definitely different in the deciphered text compared to the initial plaintext.

2. Majordomo is a program that manages Internet mailing lists. If you send a message to `majordomo@foodplus.com` saying `subscribe recipes`, Majordomo will add you to the `recipes` mailing list, and you will receive several interesting recipes by e-mail every day.

   It is easy to forge mail. You can subscribe a victim, let's say `God@heaven.af.mil`, to the `recipes` mailing list, and thousands more mailing lists, by sending fake subscription requests to Majordomo. `God@heaven.af.mil` will then be flooded with mail.

   Majordomo 1.94, released in October 1996, attempts to protect subscribers as follows. After it receives your subscription request, it sends you a confirmation number. To complete your subscription, you must send a second request containing the confirmation number.

   Majordomo 1.94 generates confirmation numbers as follows. There is a function h that changes strings to numbers. The `recipes` mailing list has a secret string k. The confirmation number for an address a is $h(ka)$. For example, if the secret string is `ossifrage`, and the address is `God@heaven.af.mil`, the confirmation number is $h($`ossifrageGod@heaven.af.mil`$)$.

   The function h produces a 32-bit result, computed as follows. Start with 0. Add the first byte of the string. Rotate left 4 bits. Add the next byte of the string. Rotate left 4 bits. Continue adding and rotating until the end of the string.

   Explain how to subscribe `God@heaven.af.mil` to the `recipes` mailing list despite this protection, and explain what Majordomo 1.94 should have done.

3. Here is a toy version of a Wegman-Carter message authentication with which A and B can authenticate t messages: Fix a prime p, e.g. $p = 1000003$. Randomly generate integers $r, s_1, s_2, \ldots, s_t \in \{0, 1, 2, \ldots 1000002\}$. These values are the shared secrets; r is the overall secret and the s_i are per message secrets.

To authenticate the i-th message m_i the sender expresses m_i in base p as $m_i = m_{i,0} + m_{i,1}p + m_{i,2}p^2 + ... + m_{i,n}p^n$ and computes the authenticator as

$$a = m_{i,0}r + m_{i,1}r^2 + m_{i,2}r^3 + ... + m_{i,n}r^{n+1} + s_i \bmod p.$$

For simplicity we will do $i = 1$ and omit the extra indices. Compute the authenticator for $m = 4543565424359792834759284 37, r = 483754, s = 342534$.