Mathematical Cryptology, 2014

Dept. of EIT
Lund University

WEEK 3,

Reading instructions:

- Chapter 6
- Chapter 7, 8, 9, 10 (repetition, not lectured)
- Chapter 11

There will be 4 hours of lectures covering the above.

Hand-in exercises:

1) Review exercises:

2) Standard exercises:

3) Programming exercise: 5.12 (maximum 5 hours)

Hand in no later than **February 19, 2014.**

Ch 5: Programming exercise:
5.8 Implement the LLL algorithm and check that it works for some simple case from the book, for example page 307. Try to use it to solve the knapsack problem given at the end of this sheet. Read about the knapsack problem in the book, p. 349-. (You should spend no more than 10 hours to complete this project)

Ch 6: Review exercises:
6.1 What is the maximum and average number of multiplications in the basic binary exponentiation algorithm?
6.2 Why is the smallest possible Hamming weight of an RSA public exponent equal to two?
6.3 What stops one using signed exponentiation techniquest in an RSA implementation?
6.4 Why is Karatsuba multiplication rarely used in an RSA implementation?
6.5 What is meant by Montgomery representation and why does it offer advantages in systems implementing algorithms such as RSA?
6.6 In algorithms for finite fields of characteristic two, why does one use an analogue of Karastsuba multiplication and not Montgomery multiplication?

Ch 6: Standard exercises:
6.7 Assume modular multiplication of a k-bit number requires k^2 operations. How much faster is two 512 512-bit modular exponentiations by 512-bit exponents, compared to a single 1024-bit modular exponentiation by a 1024-bit exponent? Conclude that the CRT method for RSA decryption and signing is more efficient.

6.8 Suppose a device uses CRT to speed up RSA decryption or signatures, i.e. it computes

$$m_p = c^{d \pmod{p-1}} \pmod{p},$$

$$m_q = c^{d \pmod{q-1}} \pmod{q},$$

and then computes m from m_p and m_q cia CRT. However, an attacker manages to get the device to compute m_p incorrectly. Show that if m_q is still computed correctly then the attacker can use this broken device to recover the private key.
6.9 Show that the RSA encryption and decryption algorithms can be implemented in $O(n^3)$-bit operations, where n is the bit length of the modulus N.
6.10 Show that ElGamal encryption requires about $2 \log p$ multiplications modulo p. Hence deduce its bit complexity is $O((\log p)^3)$.

Ch 7: Review exercises:
7.1 What are the three most common letters in English? What are the most common bigrams and trigrams in English?
7.2 Explain how the VigenÃľre cipher is related to the shift cipher.
7.3 Explain relations between the substitution cipher and the Enigma cipher.
7.4 Describe the difference and similarities between permutation and substitution as cipher components.

Ch 7: Standard exercise:
7.5 In a chosen plaintext attack the attacker is allowed to ask for the encryption of a plaintext of her choosing. Show that the Caesar, VigenÃľre and substitution ciphers can be broken instantly under a chosen plaintext attack. Determine the smallest amount of plaintext needed.

Ch 9: Review exercises:
9.1 Is an attacker more interested in $p(C = c|P = m)$ or $p(P = m|C = c)$? Explain.
9.2 How is entropy defined?

9.3 If a random variable X takes at most t values, what are the maximum and minimum values possible for $H(X)$?
9.4 Are $H(P|K,C) = 0$ and/or $H(K,P) = H(K) + H(P)$ always true for any cipher? Explain why/why not.
9.5 Define the terms spurious keys and unicity distance.

Ch 11: Review exercises:
11.1 What is meant by semantic security, and how does this relate to perfect security?
11.2 Give the definition of polynomial security.
11.3 What restriction is put on the decryption oracle in an adaptive chosen ciphertext attack?
11.4 Give two problems with the RSA algorithm as described earlier.

A Appendix to programming exercise

Which subset of the following 31 numbers sum to 54145346795 ?

The numbers are 864197523, 1728395046, 3580246881, 7407407340, 6210443074, 3939971331, 8497226607, 8760451975, 435617527, 1858889366, 4828889833, 1917605583, 5193235845, 2893211185, 7391360627, 7536374327, 7949858516, 8900283683, 1949848833, 6121919868, 5614776754, 4723947315, 3065745226, 8847539810, 2585101821, 8133166578, 1403268935, 6016414384, 6391420098, 7264888315, 160539565

(This exercise has not been checked carefully, so if you run into problems, get help!)