CHALMERS TEKNISKA HÖGSKOLA
Datavetenskap
Katerina Mitrokotsa DIT250/TDA351

Exam in Cryptography

Tuesday January 13, 2013, 14:00 – 18.00.
Teacher: Katerina Mitrokotsa, phone 076 200 11 68.
Tillåtna hjälpmedel: Typgodkänd räknare. Annan minnestömd räknare får användas efter godkännande av kursansvarig vid dennes besök i skrivsalen.
Allowed aids: Approved calculator. Other calculators with cleared memory may be used after approval by the responsible teacher.

The exam has 6 problems with a total of 50 points. 22/31/40 points are needed for grade 3/4/5.

Answers must be given in English and should be clearly justified.

1. (a) How can we sign and how can we verify a signed message using textbook RSA? (2 p)

   (b) Bob has received from Alice two signed documents (m_1, s_1) and (m_2, s_2). What problem does this cause and how it can be avoided? *Hint:* Can Bob generate a new signed message? (2 p)

   (c) Describe how the *meet-in-the-middle* attack works against textbook RSA. (4 p)

   (d) How do we define Chosen Ciphertext security of a public key encryption scheme E (i.e. IND-CCA definition). *Hint:* When does the adversary win or break E against CCA attacks (use a figure to describe your answer)? (5 p)

   (e) Is textbook RSA secure against CCA? Justify your answer. (3 p)

   (f) Is the textbook RSA signature scheme secure against CCA? Justify your answer. (3 p)

2. We consider Triple DES encryption, in the common form

$$E3_{(K_1,K_2)}(B) = E_{K_1}(D_{K_2}(E_{K_1}(B)))$$

   where E_K and D_K denote the standard (single) DES encryption and decryption functions, respectively and $E3_{(K_1,K_2)}$ denotes Triple DES encryption with key (K_1, K_2). This form of 3DES uses two keys and three operations and achieves 112 bits of security.

   A similar construction is 2DES:

$$2DES_{k_1,k_2}(m) = E_{k_1}(E_{k_2}(m))$$

   However 2DES does not achieve 112 bits of security, due to the *meet-in-the-middle* attack.

   (a) Describe the steps of the meet-in-the-middle attack in detail, if necessary use also a figure. (4 p)

(b) What level of security does 2DES achieve (i.e., how many steps of computation the adversary has to do for the attack) (1 p)?

3. We consider Elgamal encryption using a generator g for $\mathbb{Z}_p^*$ for some large prime p. Remainder: Every user chooses a random private key $x < p$ and computes the public key $X = g^x$. To encrypt message m for a user with public key X, the sender chooses a random $y < p$ and computes the encryption $(g^y, m \cdot X^y)$.

   (a) Describe how decryption is done. (2 p)

   (b) How is Elgamal encryption related to the Diffie-Hellman key exchange protocol? Describe the Diffie-Hellman (DH) protocol in detail. (2 p)

   (c) Why is it insecure against man-in-the-middle (MiM) attacks? Describe the MiM attack against the DH protocol in detail. (2 p)

   (d) Consider an improvement of the DH protocol (i.e. the MTI/A0 protocol) in which Alice and Bob have both chosen long-term keys $A = g^a$ and $B = g^b$, respectively, and certificates for these.

   Below x and y are chosen at random during protocol execution.

   $$\begin{array}{lll} 1. & \text{Alice} \to \text{Bob} & : \; X = g^x, \text{Cert}(\text{Alice}, A) \\ 2. & \text{Bob} \to \text{Alice} & : \; Y = g^y, \text{Cert}(\text{Bob}, B) \end{array}$$

   Which common key could Alice and Bob compute based on B, Y and A, X respectively (2 p)

   (e) An attacker could cause a possible problem in the communication between Alice and Bob. What is this attack/problem (describe in detail)? (4 p)

4. (a) Describe the essential properties we want a cryptographic hash function to have. (2 p)

   (b) Explain briefly the birthday attack against a hash function. How does the birthday attack affect the security of a hash function i.e. what level of security does an n-bit hash function provide. (2 p)

5. We consider a protocol where Peggy proves her identity to Victor by giving evidence that she knows a secret x.

   The system involves a trusted third party T. Initially, T chooses primes p and q as in RSA and computes $N = p \cdot q$ and a RSA key pair (e, d). N and e are made public and can be used by a whole community of Peggies and Victors. T keeps the private key d for himself. All computations below are in $\mathbb{Z}_N^*$.

Whenever (a new) Peggy wants to use the system, she chooses a public key $X \in \mathbb{Z}_N^*$ (which could be based on her name, email address etc, using some public way of transforming this to a number in $\mathbb{Z}_N^*$). She sends X to T, who computes Peggy's secret key $x = X^{-d}$ and sends it to her in some secure way. Peggy then announces her public key X.

When Peggy wants to identify herself to Victor, the following protocol is used:

1. Commitment: Peggy chooses a random $r \in \mathbb{Z}_N^*$, computes $R = r^e$ and sends R to Victor.
2. Challenge: Victor chooses a random c with $1 \leq c \leq e$ and sends c to Peggy.
3. Response: Peggy computes $y = r \cdot x^c$ and sends y to Victor.

Victor now checks that $y \neq 0$ and $R = y^e \cdot X^c$; if this holds he believes that the other party is Peggy.

(a) Show that a true Peggy, following the protocol, will be identified correctly by Victor. (2 p)

(b) What is the probability of success of an attacker that tries to impersonate Peggy and does not know x? (3 p)

6. (a) What is the main advantage of one time pad and why is it hard to use in practice? (1 p)

   (b) How can we improve the practicality of one time pad? (1 p)

   (c) How do we define perfect secrecy of a cipher (E, D) over the sets $\mathcal{M}, \mathcal{K}, \mathcal{C}$ where $\mathcal{M}$ denotes the set of messages, $\mathcal{K}$ denotes the set of keys and $\mathcal{C}$ denotes the set of ciphertexts? (2 p)

   (d) Are you aware of any stream cipher that provides perfect secrecy? (1 p)