CHALMERS TEKNISKA HÖGSKOLA
Datavetenskap
Daniel Hedin DIT250/TDA351

Exam in Cryptography

Tuesday, December 17, 2013, 14.00 – 18.00.
Teacher: Daniel Hedin, phone 0709 261771.
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 7 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.

Hash functions (7 p)

1. Explain what a cryptographic hash function is and the notion of collision resistance. (3 p)

2. Explain the birthday problem and how it affects collision resistance. (4 p)

Public key (10 p)

1. Textbook RSA is multiplicatively homomorphic, i.e.,

$$RSA_{pk}(m_1) \cdot RSA_{pk}(m_2) = RSA_{pk}(m_1 \cdot m_2) \tag{1}$$

   Define how RSA encryption is done, and show that RSA is multiplicatively homomorphic, i.e., that equation 1 holds. (3 p)

2. Explain how textbook RSA can be used for digital signatures, i.e., explain how to sign and how to verify the signature. (3 p)

3. Signatures based only on textbook RSA are not secure due to RSA being multiplicatively homomorphic. Show an existential forgery attack on textbook RSA signatures, i.e., given messages, m_i, and the corresponding signatures, σ_i, show how to construct a new message m and a corresponding valid signature σ without having access to the key. (2 p)

4. How can we change the way we produce RSA signatures to protect against the above existential forgery attack? Give the construction and show that the attack no longer works. (2 p)

Block Ciphers (9 p)

1. We have argued that the construction of block ciphers is guided by the notions of diffusion and confusion. Explain the notions and how they relate to plain text recovery and key recovery attacks. (2 p)

2. As you remember, a Feistel network is a common way of constructing block ciphers described by

$$L_{i+1} = R_i$$
$$R_{i+1} = L_i \oplus f(R_i, K_i)$$

   What properties do we expect the round function f to have? In particular, does f have to be invertible? Justify your answers. (3 p)

3. DES is an old block cipher that should no longer be used primarily due to its small key size of 56 bit, which makes brute force attacks possible. We have seen that the key size for DES can be increased by repeated encryption under different keys. For instance, 3DES uses two keys and three operations to achieve 112 bits of security as follows

$$3DES_{k_1,k_2}(m) = E_{k_1}(D_{k_2}(E_{k_1}(m)))$$

   where E_k is DES encryption using key k and D_k is DES decryption using key k.

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

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

   Explain how the meet-in-the-middle attack works and what security level 2DES achieves, i.e., how many steps of computation the adversary has to do for the attack. (4 p)

Protocols (7 p) the ISO 9798-2 protocol

Recall the ISO 9798-2 protocol:

$$\begin{array}{lll} 1. & A \to B & : \quad A, N_A \\ 2. & B \to A & : \quad \{A, N_A, N_B\}_{K_{AB}} \\ 3. & A \to B & : \quad \{N_B, N_A\}_{K_{AB}} \end{array}$$

1. What does ISO 9798-2 achieve? Give a detailed explanation of why it achieves this. (3 p)

2. Assume that we want to realize $\{\cdot\}_{K_{AB}}$ using a block cipher and that the size of each part of the message A, N_A, and N_B is equal to the block size. What block cipher mode could be used for the encryption? (2 p)

3. Consider the following change to the third message of the protocol.

   $$3. \quad A \to B \quad : \quad \{N_A, N_B\}_{K_{AB}}$$

   Are you still able to securely use the mode you suggested above? You must explain your answer; a simple yes or no answer does not suffice. (2 p)

Protocols (5 p) the Schnorr protocol

Setting: Group of prime order q with generator g. P has private key x and public key $X = g^x$.

- P generates random $r \in \mathbb{Z}_q$, computes $R = g^r$ and sends R to V.
- V generates random $c \in \mathbb{Z}_q$ and sends it to P.
- P computes $z = c \cdot x + r$ and sends it to V.

After this sequence of messages V checks if $g^z = R \cdot X^c$.

1. Justify why this protocol works. (2 p)

2. Show that if an eavesdropper is able to get two runs of the protocol that are using the same commitment r the eavesdropper can compute the secret x, i.e., show how to compute x from (R, c_1, z_1) and (R, c_2, z_2). (3 p)

Random numbers (5 p)

1. Justify why block ciphers, MACs and cryptographic hash functions can be used to generate random numbers. (2 p)

2. Show how you can use the counter mode (CTR) to create a pseudo random number generator $prng(x)$, where x is the seed. What is the period for your generator, i.e., how many bits can it output before repeating? (3 p)

Security notions. (7 p) We recall the notion of *indistinguishability under chosen plaintext*, IND-CPA. Let $(\mathcal{K}, E, D)$ be a public key system with key generation algorithm $\mathcal{K}$, encryption aglorithm E and decryption algorithm D all known by the adversary. We say that the public key system satisfies IND-CPA if the attacker has *negligible probability* of winning the following game.

$$
\begin{aligned}
&\text{IND-CPA}: \\
&\quad (pk, sk) \leftarrow \mathcal{K}() \\
&\quad m_0, m_1 \leftarrow \mathcal{A}_1(pk) \\
&\quad b \xleftarrow{\$} \{0,1\} \\
&\quad c \leftarrow E(pk, m_b) \\
&\quad b' \leftarrow \mathcal{A}_2(pk, c)
\end{aligned}
$$

The game is read as follows:

- We generate fresh keys.
- The adversary is allowed to chose messages freely and encrypt them using the public key. Once he is finished he selects two plaintexts.
- We throw a fair coin and depending on the outcome encrypt one of the plaintexts selected by the adversary, and give the adversary the resulting ciphertext.
- To win the adversary should decide which of the plaintexts we encrypted.

1. Give an explanation for why this is a good security notion. (2 p)

2. Show that no deterministic cipher can satisfy IND-CPA, i.e., assuming that E is deterministic, create an adversary $A = (A_1, A_2)$ that has non-negligible probability to win the game. (4 p)

3. What is the probability that your adversary wins? (1 p)