CHALMERS TEKNISKA HÖGSKOLA
Datavetenskap
Björn von Sydow DIT250/TDA351

Exam in Cryptography

Thursday December 20, 2012, 14.00 – 18.00.
Teacher: Björn von Sydow, phone 0722 39 14 01.
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.

1. Explain briefly similarities and differences between cryptographic hash functions and message authentication codes (MAC's). (3 p)

2. We recall the two modes of operation CBC and CTR for a block cipher E:

CBC	CTR
$C_0 = IV$	$C_0 = IV$
$C_i = E_K(M_i \oplus C_{i-1})$	$C_i = M_i \oplus E_K(IV\|\|i)$

   (a) Describe for both modes how decryption is done. (2 p)

   (b) Alice wants to send encrypted messages to Bob, with whom she shares an AES key, but her encryption software runs on a computer which she accesses over a local but insecure network. She therefore decides to blind her messages on her computer, by randomly choosing a block N and xoring every block of her message with N, before sending it for encryption.[1] However, she also wants to unblind the encrypted message that she gets back, before sending it to Bob, so that Bob need not be aware of the blinding and can just decrypt it in the standard way.

   Encryption can be done in either CBC or CTR mode. Help Alice to choose one of the two modes and explain to her how to do the unblinding. (3 p)

   (c) The system described above works well, but Alice now considers improving her blinding scheme, as follows:

$$
\begin{aligned}
M'_1 &= M_1 \oplus N \\
M'_i &= M_i \oplus M'_{i-1}, \quad i > 1.
\end{aligned}
$$

   The blinded message is $M'_1 M'_2 \ldots$. Alice's reasoning is that each message block will now be xor-ed not only with N, but also with all previous message blocks, and thus make unblinding without knowing N more difficult. Do you agree? (2 p)

[1] **Note:** This is *not* a recommended practice, but a dubious procedure cooked up just for this exam.

3. Alice regularly gets contracts, prepared by her business contacts, to sign digitally. Her practice is to always add a space somewhere in the contract, before hashing and signing. She then returns the slightly modified contract along with her signature.

   Explain why this practice of modifying the received contract is a good idea. (4 p)

4. We consider stream ciphers, where a plaintext m is encrypted by bitwise xor-ing with an equally long keystream k, i.e. $c = m \oplus k$. Different ways of generating the keystream can give ciphers with very different properties. Discuss briefly advantages and disadvantages of the ciphers resulting from the following two choices.

   (a) k is a bitstring, which is generated from a truly random source and used for only one encryption. (3 p)

   (b) k is generated by a linear feedback shift register (an LFSR), of size 40 bits, initialized with a 40 bit secret key, which is used for an extended period of time. (3 p)

5. We consider the following scheme for signcryption, i.e. combined encryption and signing of a message.

   The setting is a cyclic group G of prime order q with generator g. Each user has a private key $x \in \mathbb{Z}_q$ and a public key $X = g^x$. The system also uses a hash function H and a secret key encryption method E; E_K denotes encryption and D_K decryption with key K.

   Assume that Alice has private key x_A and public key X_A and that Bob similarly has keys x_B and X_B. To encrypt and sign message m for Bob, Alice does the following:

   - Choose a random integer $z < q$.
   - Compute $K = X_B^z$.
   - Compute $c = E_K(m)$.
   - Compute $r = H(m||X_A||X_B||K)$.
   - If $r + x_A = 0 \in \mathbb{Z}_q$ then start again, choosing a new z, otherwise compute $s = z \cdot (r + x_A)^{-1} \in \mathbb{Z}_q^*$.
   - The signcrypted message is (c, r, s).

   On receipt of (c, r, s), Bob does the following:

   - Compute $K = (X_A \cdot g^r)^{s \cdot x_B}$.
   - Compute $m = D_K(c)$.
   - If $H(m||X_A||X_B||K) = r$, then m is accepted as the message signed by Alice; otherwise the message is rejected.

   Motivate in detail why a correctly signcrypted message by Alice will be correctly decrypted and verified by Bob. (6 p)

6. Tsudik and Herreweghen proposed the following protocol in which A and B use a MAC algorithm, and a long-term shared key K_{AB} for this MAC, to agree on a session key K_S:

$$\begin{array}{lll} 1.\ A \to B & : & A, N_A \\ 2.\ B \to A & : & N_B, \mathrm{MAC}_{K_{AB}}(N_A||N_B||B) \oplus K_S \end{array}$$

Here N_A and N_B are nonces chosen by A and B respectively.

(a) How does A compute the session key? (2 p)

(b) Explain for both A and B why they believe after a run that the session key K_S is fresh, i.e. that they are not subject to a replay attack which establishes some old session key. (2 p)

(c) Naive Ned argues that since B chooses a new random key K_S in message 2, his nonce N_B is not necessary and the protocol could be simplified to

$$\begin{array}{lll} 1.\ A \to B & : & A, N_A \\ 2.\ B \to A & : & \mathrm{MAC}_{K_{AB}}(N_A||B) \oplus K_S \end{array}$$

Show that Ned is wrong by demonstrating how an adversary, who records a run of this simplified protocol and also gets hold of the session key for this run, can later agree on keys with Bob at will. (3 p)

(d) The protocol in (a) has the disadvantage that the key is chosen by B alone. In general, it is preferable to use a trusted third party for key selection. Give two reasons why this is better. (2 p)

(e) Another option is to allow both parties to contribute to the common session key. Describe the basic Diffie-Hellman protocol, which uses this policy. What is the main weakness with this protocol? (4 p)

7. In this problem we consider Manger's attack on RSA encryption. Assume an RSA user, who uses a modulus $N (= p \cdot q)$ of length 1024 bits = 128 bytes, with public key e and private key d.

The user employs an unspecified padding system, where a plaintext m to be encrypted must be much shorter than 128 bytes, say 50 bytes long, and is padded to a bitstring m_p of length 128 bytes, which is then seen as integer in $\mathbb{Z}_N^*$ and encrypted with textbook RSA to give ciphertext $c = m_p^e$. Before encryption, one should check that $m_p < N$. A cheap way to guarantee this is to require that the most significant byte (i.e. the eight most significant bits) of m_p is zero, so that in fact $m_p < 2^{1016}$. We introduce the notation B for 2^{1016}, so padding has the property that $m_p < B$.

Decryption of a ciphertext c now consists of several steps:

1. Do textbook RSA decryption to get m_p, i.e. compute c^d.
2. Check that most significant byte of m_p is 0; if not, decryption fails.
3. Remove padding to recover m or, if padding is wrong, note that decryption fails.

We consider a scenario where the adversary is able to mount a chosen ciphertext attack, so he can construct and send ciphertexts to the user, who will decrypt them. With a good padding system, almost all ciphertexts constructed by the adversary will lead to decryption failure, but we assume that the adversary can distinguish between failure in step 2 and failure in step 3. Possible ways to achieve this is through measuring time for decryption, or from the user giving different error messages in the two cases or possibly for other reasons.

Assume now that the adversary has a ciphertext c, which is a proper encryption of some plaintext m. We will now study an attack, by which the adversary can recover m_p (and then himself remove padding to get m).

(a) The adversary knows that $m_p < B$ (since c is the result of a correct encryption). His first step is to construct a ciphertext c_1, which is the textbook encryption of $2 \cdot m_p \in \mathbb{Z}_N^*$. How does he construct c_1? Explain! (3 p)

(b) Assume that the response from the user, when she receives c_1, is that step 2 in the decryption fails. Now what does the adversary know about m_p? (2 p)

(c) Next, the adversary wants to construct c_2, which is the textbook encryption of $(k+1) \cdot m_p \in \mathbb{Z}_N^*$, where k is the eight most significant bits of N, seen as an integer. How does he construct c_2? (This is no trick; if you could do (a), you can do this.) (1 p)

(d) Assume now that decryption fails only in step 3. What does the adversary now know about m_p? (3 p)

(e) Explain some measure that a padding system can use in order to achieve the above-mentioned effect that almost all the adversary's ciphertexts lead to decryption failure. (2 p)

We leave the attack here; the interested reader may be able to see how the adversary can proceed to do a kind of binary search to determine m_p completely in around 1100 queries.