T-79.4501 Cryptography and Data Security
EXAM
Tuesday, October 28, 2008
SOLUTIONS

1. Let us denote the encryption key

$$\begin{pmatrix} x^2 & 1 \\ 1 & x+1 \end{pmatrix}$$

by K.

a) (3 pts)

$$K \begin{pmatrix} x^2 \\ x+1 \end{pmatrix} = \begin{pmatrix} x^4+x+1 \\ x^2+(x+1)^2 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

b) (3 pts) The task is to find the inverse matrix K^{-1} of K such that $KK^{-1} = K^{-1}K = I$ where I is the 2×2 identity matrix. By matrix algebra

$$K^{-1} = (\det K)^{-1} K^T = (\det K)^{-1} \begin{pmatrix} x+1 & 1 \\ 1 & x^2 \end{pmatrix}.$$

Now we have at least the following two possibilities to continue:

(1) We have $\det K = x^3+x^2+1$ and can obtain its inverse x^2 using the Extended Euclidean algorithm.

(2) By a) we know that the second column of K^{-1} is $(x^2, x+1)^T$. Hence the multiplier $(\det K)^{-1}$ must be equal to x^2.

We get

$$K^{-1} = \begin{pmatrix} x^3+x^2 & x^2 \\ x^2 & x+1 \end{pmatrix}.$$

2. In CBC encryption, $C_i = E_K(P_i \oplus C_{i-1})$, for all $i = 1, 2, \ldots, t$, where $C_0 = IV$. Then decryption is computed as $P_i = D_K(C_i) \oplus C_{i-1}$. Error occurs in exactly one C_j, where $0 \leq j < t$. We denote the erroneous ciphertext by C'_j.

a) (3 pts) Exactly two decryptions depend on C'_j:

$$\begin{aligned} P'_j &= D_K(C'_j) \oplus C_{j-1} \\ P'_{j+1} &= D_K(C_{j+1}) \oplus C'_j, \end{aligned}$$

where we have denoted the erroneous decryptions by P'_j and P'_{j+1} .

b) (3 pts) The differences in bits between the original plaintexts and the received erroneous plaintexts are:

$$\begin{aligned} P_j \oplus P'_j &= D_K(C_j) \oplus C_{j-1} \oplus D_K(C'_j) \oplus C_{j-1} = D_K(C_j) \oplus D_K(C'_j) \text{ and} \\ P_{j+1} \oplus P'_{j+1} &= D_K(C_{j+1}) \oplus C_j \oplus D_K(C_{j+1}) \oplus C'_j = C_j \oplus C'_j. \end{aligned}$$

Hence the error in block P_j looks random assuming that D_K is the decryoption of a strong block cipher. The error in P_{j+1} is exactly in those bits that are erroneous in C'_j.

3. a) To use the Chinese Remainder Theorem we denote as usual $m_1 = 3$, $m_2 = 5$, $m_3 = 7$, $M = m_1 m_2 m_3 = 105$, and

$$\begin{aligned} M_1 &= m_2 m_3 = 35 \\ M_2 &= m_1 m_3 = 21 \\ M_3 &= m_1 m_2 = 15. \end{aligned}$$

Then

$$\begin{aligned} u_1 = M_1^{-1} \bmod m_1 &= 35^{-1} \bmod 3 = 2^{-1} \bmod 3 = 2 \\ u_2 = M_2^{-1} \bmod m_2 &= 21^{-1} \bmod 5 = 1^{-1} \bmod 5 = 1 \\ u_3 = M_3^{-1} \bmod m_3 &= 15^{-1} \bmod 7 = 1^{-1} \bmod 7 = 1. \end{aligned}$$

Then

$$x = \sum_{i=1}^{3} x_i u_i M_i = 2 \cdot 2 \cdot 35 + 3 \cdot 1 \cdot 21 + 2 \cdot 1 \cdot 15 = 233 \equiv 23 \pmod{105}.$$

b) The RSA modulus is $n = 105 = 3 \cdot 5 \cdot 7$. We compute $\phi(n) = 2 \cdot 4 \cdot 6 = 48$. Then

$$d = e^{-1} \bmod \phi(n) = 7^{-1} \bmod 48 = 7,$$

as $7 \cdot 7 = 49 \equiv 1 \pmod{48}$.

4. (6 pts) For the Man-in-the-Middle Attack on the basic (unauthenticated) Diffie-Hellman key exchange protocol see Lecture 9 Slide 10. The small subgroup attack discussed in Homework 5 Problem 4 can also be considered as a Man-in-the-Middle Attack, since it involves an active attacker, who modifies the messages. It differs from the basic Man-in-the-Middle Attack in two aspects. First, it is only possible if the group has a small subgroup, where the attacker can force the final key to belong. Secondly, it can be applied also to the authenticated Diffie-Hellman key exchange.

5. (6 pts) The 64-bit output Y from the counter mode PRNG is computed as $Y = E_K(X)$ where X is a 64-bit counter value. We denote the initial counter value by X_0. Starting from this value, the counter X will run through all 64-bit values until it takes value X_0 again. Encryption of 64-bit blocks using Triple DES, which is a 64-bit block cipher, is bijective. Therefore the output Y from the PRNG will run through all 64-bit numbers, that is, 2^{64} different values without repetition, until it takes value $E_K(X_0)$ again.

   The true random number generator selects each 64-bit output uniformly at random from all possible 2^{64} outputs. Hence at each time it is there is a chance that a previously selected value is selected again.

   Therefore, is a repetition occurs in a sequence generated by a black box generator before 2^{64} numbers have been generated, we know that the black box contains a true random number generator.

   By Birthday Paradox, the probability that a repetition is detected after $\sqrt{2^{64}} = 2^{32}$ values have been generated is about $1/2 = 50\%$.