HW.No .2 **chapter 3 and 4** **Due time: 1401.1.15**

1. Solve the following exercises in your textbook " Understanding Cryptography": Exercise 3.1, 3.3, 3.5, 3.7and 4.9.

2. Answer the following questions:
   a. What is the difference between a block cipher and a stream cipher?
   b. What is the purpose of the S-boxes in DES?
   c. Show that DES decryption is, in fact, the inverse of DES encryption.
   d. Check and compare the security of AES and DES in terms of attacks Brute-Force Attack, Statistical Attacks and Differential and linear Attacks.
   e. Compare AES to DES. For each of the following elements of DES, indicate the comparable element in AES or explain why it is not needed in AES.
      i. XOR of subkey material with the input to the f function
      ii. XOR of the f function output with the left half of the block
      iii. The f function
      iv. Permutation P
      v. Swapping of halves of the block

3. This problem provides a numerical example of encryption using a one-round version of DES. We start with the same bit pattern for the key K and the plaintext, namely: in hexadecimal notation for key and plaintext: 0 1 2 3 4 5 6 7 8 9 A B C D E F

in binary notation: 0000 0001 0010 0011 0100 0101 0110 0111 1000 1001 1010 1011 0100 1101 1110 1111.

a. Derive K1, the first-round subkey.
b. Derive L0, R0.
c. Expand R0 to get E[R0], where E[·] is the expansion permutation
d. Calculate A = E[R0] ⊕K
e. Group the 48-bit result of (d) into sets of 6 bits and evaluate the corresponding S-box substitutions.
f. Concatenate the results of (e) to get a 32-bit result, B.
g. Apply the permutation to get P(B).
h. Calculate R1 = P(B)⊕ L0.
i. Write down the ciphertext.

4. Find all irreducible polynomials of degree 4 over GF (2).

5. The 32-bit swap after the sixteenth iteration of the DES algorithm is needed to make the encryption process invertible by simply running the ciphertext back through the algorithm with the key order reversed. This was demonstrated in Problem 3.7. However, it still may not be entirely clear why the 32-bit swap is needed. To demonstrate why, solve the following exercises.

First, some notation: A||B = the concatenation of the bit strings A and B

Ti (R||L) = the transformation defined by the ith iteration of the encryption algorithm.

TDi (R||L) = the transformation defined by the ith iteration of the decryption algorithm.

T17(R||L) = L||R. This transformation occurs after the sixteenth iteration of the encryption algorithm.

a. Show that the composition TD1(IP(IP-1(T17(T16(L15||R15))))) is equivalent to the transformation that interchanges the 32-bit halves, L15 and R15. That is, show that

TD1(IP(IP-1(T17(T16(L15||R15))))) = R15||L15

b. Now suppose that we did away with the final 32-bit swap in the encryption algorithm. Then we would want the following equality to hold:

TD1(IP(IP-1(T16(L15||R15))))) = R15||L15 Does it?

6. Let K =111... I l l be the DES key consisting of all 1 s . Show that if $E_K(P) = C$, then $E_K(C) = P$, so encryption twice with this key returns the plaintext.

7. Given the plaintext {00010203040506070809 0A0B0C0D0E0F} and the key {01010101010101010101010101010101},

a. Show the original contents of State, displayed as a 4 x 4 matrix.
b. Show the value of State after initial AddRoundKey.
c. Show the value of State after SubBytes.
d. Show the value of State after ShiftRows.
e. Show the value of State after MixColumns.

8. *Note: Answer the below questions making use of CrypTool; use the ECB mode for all the exercises related to the DES algorithm.*

- **Encryption/Decryption:**
  1. Search about weak keys of DES and answer the following questions;
     - i Encrypt your text twice, with DES algorithm using one of the weak keys for both rounds.
     - ii Again, encrypt the text twice, using a DES semi-weak key pair.
  2. A more secure alternative to DES is Triple DES, answer the following questions surrounding this algorithm;
     - i Why is it more secure compared to DES?
     - ii When implementing a brute-force attack, how large is its keyspace? Why is it that long?
     - iii Compare its two versions with each other.
     - iv Encrypt your text using the CrypTool Triple DES encryption scheme with your desired key.

v Knowing that CrypTool uses the second version of 3DES, and encrypts the text with k1=k3 (it uses the key of the first encryption round for both the first and the third rounds), Encrypt the same text 3 times using the simple DES algorithm, with k1 = the first half of your key in the previous part, and k2 = its second half.

vi Try to Decrypt the last output of the previous part using the Triple-DES decryption and the same key as in part "iv".

3 Another secure approach compared to DES is DESX, which makes use of key whitening, answer the following questions about this method.

i Encrypt your text using the CrypTool DESX algorithm with your desired key. (hint: this algorithm is found on Further algorithms menu)

ii Again, encrypt the text making use of simple DES and classic XOR algorithms, with k = the first 8 bytes of your previous key, k1 = its second and k2 = its third 8 bytes.

4 Encrypt your text using the AES algorithm and your desired key.

- **Analysis:**

5 Answer the below parts using CrypTool analysis tool for AES;

i Try to find the plain text utilizing the ciphertext in exercise4 without entering any parts of the key. According to the analysis time shown in standard form, calculate how long it takes the software to verify a key, and provide the answer in microseconds.

ii Now, enter 14 bytes of your key and see the ultimate results. What do you think the entropy is, and how is it related to the correct decryption of your ciphertext?

OPTIONAL QUESTIONS

9. Let x_1, = 00000000, x_2 = 00000001, x_3 = 00000010, x_4 = 00000011. Let B S (x) denote the ByteSub Transformation of x. Show that BS(x1) $\oplus$ B S (x 2) = 00011111 $\neq$ 00001100 = B S(x3) $\oplus$ B S (x4). Conclude th at the ByteSub Transformation is not an affine map.

10. Suppose we modify the Feistel setup as follows. Divide the plaintext into three equal blocks: L0, M0, Ro. Let the key for the ith round be Ki and let F

be some function that produces the appropriate size output. The ith round of encryption is given by

$$Li = Ri\text{-}1, \quad Mi = Li\text{-}1, \quad Ri, = F(Ki, Ri\text{-}1) \oplus Mi\text{-}1$$

This continues for n rounds. Consider the decryption algorithm that starts with the ciphertext An, Bn, Cn and uses the algorithm

$$Ai\text{-}1=Bi, \quad Bi\text{-}1=F(Ki, Ai) \oplus Ci, \quad Ci\text{-}1= Ai.$$

This continues for n rounds, down to A0, B0, C0. Show that Ai = Li, Bi = Mi, Ci = Ri for all i, so th at the decryption algorithm returns the plaintext. (Remark: Note that the decryption algorithm is similar to the encryption algorithm, but cannot be implemented on the same machine as easily as in the case of the Feistel system.)

gelare.oudi@ec.iut.ac.ir