2 ECB

2.1 Encryption and decryption of ECB

ECB mode is one of the most basic working modes of block cipher. In this mode, the information to be processed is divided into the appropriate size group, and then each packet is independently encrypted or decrypted by using the same key. As shown in the following figures:

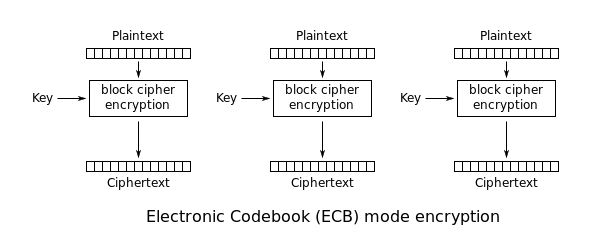
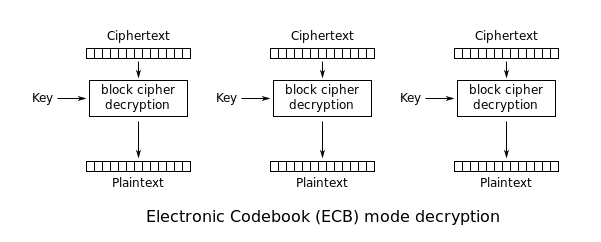


Figure 1. Encryption of ECB

Figure 2. Decryption of ECB

2.2 Implementation and problems

2.2.1Encryption

First of all, the substitution by S-box is the first step in the process of EBC encryption algorithm. Because the input plaintext is character data, the plaintext is converted into the hexadecimal number before the element is replaced. In the process of substitution, because S-box uses a byte of hexadecimal to locate a number to be replaced, substitution is completed after length (piaintext) /2 times. The second step is permutation. In this process, the plaintext is transformed into the binary number first, and then it is transformed into a 8\*8 matrix. In order to achieve the effect of permutation, suppose that row is j, column is i, let j= (i+j)%8, and then each row element moves to the correct position. If the length of the plaintext is a multiple of 8 , it can be divided into matrix 8\*8 directly, but when the length of plaintext is not multiple for 8, it needs to complement 0, let the length of plaintext%8, the remainder show the need for the number 0, using function to add 0 after plaintext and recorded the number of remainder, and than minus that when ciphertext generated to make ciphertext maintain the original length. The third process is XOR operation, generating a key matrix and transform the processed ciphertext to matrix, each element XOR with the key matrix. This is a complete process, and then, the key matrix is transformed each time after 4 rounds of operationto get the final results.

2.2.2 Decryption

In the decryption process, the ciphertext will be processed by the same but contrary operation(XOR, permutation, substitution) and we can get the original plaintext.

2.2.3 Problems

The main problems that I met is I need make sure that the length of plaintext is the multiple of 8, but the plaintext is input by random, and both of the permutation and XOR stage need transform the plaintext to 8\*8 matrix. Therefore, it needs a operation to add 0 to the plaintext and decide the new length of plaintext. Once complete this process, it is helpful to the later program.

2.3 Test

The plaintexe and ciphertext show below, and it can be seen that plaintext can be encrypted totally. And it recover to original plaintext when the ciphertext be decrypeted.

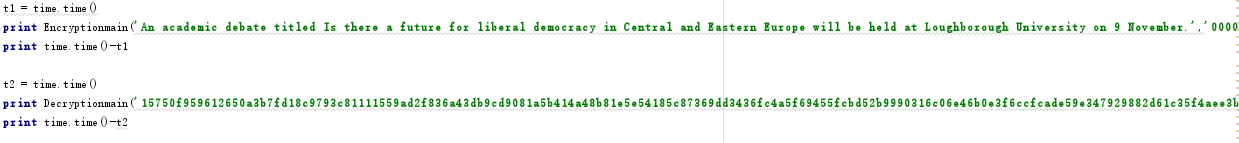


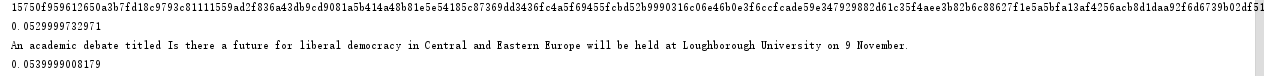
Figure3. Input

Figure4.Output

2.4 Performance disscuss

The figure 4 illustrates that generating the ciphertext take 0.0529999732971 second and the decryption take about 0.0539999008179 second, it is not to fast but actually in a acceptable range.

2.5 Analysis of the security of this cipher

As a basic working mode, ECB has the characteristics of simple operation and easy implementation. At the same time, because of the independence of the grouping, it is conducive to parallel processing, and can prevent errors.

On the other hand, because the encryption of all packets is the same, the duplicate content in plaintext will be reflected in ciphertext, so its security is not the best.

Therefore, the ECB model is only suitable for the security protection of small amount of information.