ASSIGNMENT-4

Team Name: codemeisters

Members:

VIJIT MALIK (170791)

HITESH KUMAR (170305)

PRAVEEN KUMAR (170504)

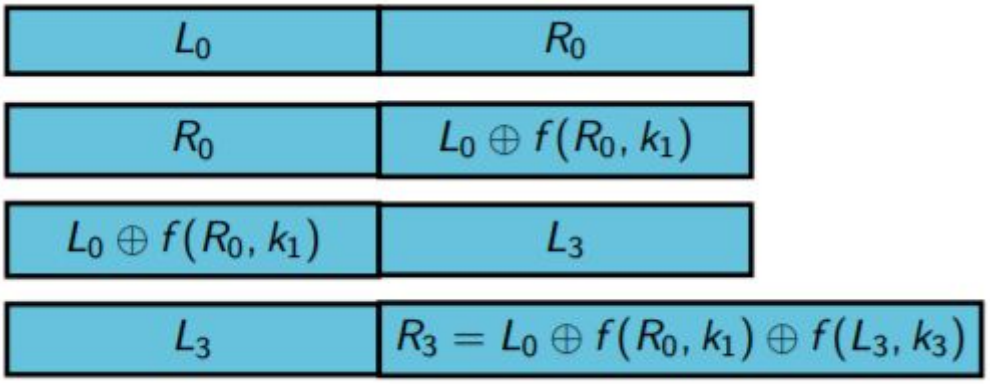
STEPS:

1. IDENTIFICATION OF THE TYPE OF ENCRYPTION.

We entered enter-dive-back-pull-go-back-back-wave and read in the command and came across the last page of chapter 4 which tells that the encryption type used is DES.

But since sir has said in the class that Assignment 4 is 3 round DES. We started breaking 3 round DES.

2. BREAKING THE 3 ROUND FEISTEL NETWORK:



To break this, we have to perform two known plaintext attacks for L_0R_0 and $L_0^*R_0^*$ with $R_0 = R_0^*$.

Then the outputs will have the relation $R_3 \oplus R_3^* = L_0 \oplus L_0^* \oplus f(L_3, k_3) \oplus f(L_3^*, k_3)$

We have $L_3 \oplus L_3^*$ and $f(L_3, k_3) \oplus f(L_3^*, k_3)$

Choose $R_0 = R_0^*$ so that $f(R_0, k_1) \oplus f(R_0^*, k_1) = 0$. Then, we can calculate $f(L_3, k_3) \oplus f(L_3^*, k_3)$

3. BREAKING THE ENCRYPTED TEXT:

We were very amazed to see that the two letters of the text represents one byte.
After trying random words of different size as input, we saw that the output contains only small letters and that too from letter ‘f’ to ‘u’. It means that we got to know how two letters of the output represents one byte.
But it wasn’t that easy! The unusual thing we noticed is that when we input words of less than 16 bytes we get output of 8 bytes and when the input is more than 16 bytes we get output of 16 bytes.
Isn’t it unusual?
But after thinking that it is not possible we thought that input should also be bounded to letters from ‘f’ to ‘u’.

For breaking this cipher text we chose two plaintext L0R0 and L0’R0’ as described above, and we have two corresponding encrypted text L3R3 and L3’R3’. It means we know XOR of input and XOR of output. Coming from output side we can get R2 as it is equal to L3. Also we can get XOR of L2 and L2’ as it is XOR of L0 and L0’. After applying permutation and expansion we come across S-box with the XOR value on the both side of S-Box.
Here we have 64 different possible input for each block of 6 bit input. So we get all possible values of input and use them to get output of S-box.
Then we find XOR of S-box output value and compare the real output of S-box.
If both are same then our possible input possible.
Then we calculate the key for 6 bit first block using possible input and real input before key.
We check these condition for all 8 blocks.
Again we give another plain text pair with same Right Part text.

We get keys from both plane text pairs. And eliminate the uncommon key values. After using some other plain text pair we find 3rd round key.
After that we find the all possible key3 with 256 keys. We have key3 and have 256 possibilities of the main 56 bit key. We brute force over all 256 possibilities for some pairs and get key1 and key2.

Now we know all 3 keys.
So we decipher the given cipher using found keys.
The plain text came out to be this for the cipher “rlllrqikiiiimmuttnigrohjmmjulqqk” corresponding to when we enter “password”:
----> “inspgjqijhrtnitiisjrtuqmoggqkjsj”

Therefore this plaintext worked and we broke the cipher.