We observed that the ciphertext uses only 16 letters from the alphabets and then after looking at various ciphertext, we observed that it uses characters only from (f to u).

These are 16 characters and we name them from 0 to 15 and consider them as new characters for plaintext and use these in all the analysis.

We are going to use 2 3 - three round characteristics to solve the DES.

characteristization 1:

[see FIg 1]

We can see that for the fourth round the input xor for S2,S5,S6,S7,S8 blocks is exactly 0. Hence output xor will be 0 with probability 1.

We will use the same procedure as told in class but we will modify it a little

[see FIg 2]

Notice we only know a few bits of R5[30 bits] with probability 1/16((¼x¼)).

For round 6:

[see FIg 3]

We take 2 cipher texts say R_1 L_1 and R_2 L_2 .

We can calculate the input to the S2 block as :

Expansion (L1)[6:12] + k_{6,2} , Expansion(L2)[6:12] + k_{6,2}

And we know th output xor of S2 :

P^{-1} (R5 XOR R1 XOR R2) [4:8] as ((R5 XOR R1 XOR R2)) is completely known;

We can check the values of the keys that satisfy this.

by using :

S2(Expansion(L1)[6:12] XOR k_{6,2}) XOR S2(Expansion(L1)[6:12] XOR k_{6,2} =

p^{-1}(R5 XOR R1 XOR R2)[4:8]

We then do this for 1000 ciphertext pairs and also not the frequency of each key that appears. Now we take the key with the highest frequency .

[DO CHECK THE CODE]

we have written the frequency also of the key we are taking .

Hence we were able to determine k_{6,2} , k_{6,5} , k_{6,6} , k_{6,7} , k_{6,7} , k_{6,8} , using this characterization.

Using the 2nd characterization:

[see FIg 4]

Hence in Round 4, input xor of S1,S2,S4,S5,S6 block are zero and corresponding output xor also will be zero.

Following the same procedure we arrive at k_6_1 and k_6_4 .

Now we know 42 bits of k_6 corresponding to S1,S2,S4,S5,S6,S7,S8.

Then we will brute force our way to get the 56 bit key.

our initial key will be of the form -
x11xx1xx01011x100xx11x11100x0010100x00110100x11x0111x001

In this key there are 14 ‘x’ denotes the unknown value and so we use 2^14 combinations and brute force our way to find the actual key which is -
01101110010111100111101110000010100100110100011001110001

Now we decrypt our password which comes out to be - ‘mhmfmmlslklglomflslgifififififif’

This is in the form where 1 character is represented by 4 bits as f=0000 (we start from this and so on).

Then we convert this encrypted text into bits and take 8 bits combined as in regular characters and convert them into integers and take the corresponding original characters as in ascii table.

The value of ‘if’ comes out to be ‘48’ which does not correspond to any character, so we think that it is padding, so we remove this.

After that we find this value - ‘rpwmeaipma’.

Then we entered this and completed level 4.

For more details, Please see the code.