Q1 Team Name

0 Points

Enigma

Q2 Commands

10 Points

List all the commands in sequence used from the start screen of this level to the end of the level. (Use -> to separate the commands)

go -> jump -> dive -> back -> pull -> back -> back -> go -> wave -> back -> back -> thrnxxtzy -> read -> 134721542097659029845273957 -> c -> read-> password-> piqxkbgghq

Q3 CryptoSystem

5 Points

What cryptosystem was used at this level? Please be precise.

6-round Data Encryption Standard(DES)(Block Cipher)

Q4 Analysis

80 Points

Knowing which cryptosystem has been used at this level, give a detailed description of the cryptanalysis used to figure out the password. (Use Latex wherever required. If your solution is not

password. (Use Latex wherever required. If your solution is not readable, you will lose marks. If necessary, the file upload option in this question must be used TO SHARE IMAGES ONLY.)

We had to first find the magic wand at the bottom of the river and bring it up to the surface. After that, we went back to level 3 and freed the spirit there. After returning back to the first screen of level 4, we typed the command 'read' in the console, following a few hints provided by the spirit, and then typed the word 'password' as instructed in the message. The ciphertext was then revealed to be **knioqsdhrkroomqmrgdrqomddilqlnimi**. We had to decrypt this ciphertext to cross-level 4 in order to complete the task.

There were several hints that the cypher at Level 4 was encrypted using the DES algorithm. It has been stated that the algorithm could be a 4, 6, or 10-round DES algorithm. But since a 4-round DES is easy to break and a 10-round DES will be quite difficult to break, and because the spirit says, "but this one certainly isn't 10-round...", the likelihood of the crypt algorithm to be 6-round DES increased. Using 6-round DES, we attempted to decipher the password by breaking down 6-round DES with the code provided in Analysis.ipynb, assuming that if it didn't work with 6-DES, it would work with 4-DES.

We used a chosen-plaintext attack to break the 6-round DES. In this attack method, for cryptanalysis, the attacker generates samples of plaintexts, gets the sender to encrypt them and then use the obtained pairs of plaintexts and ciphertexts to find the key used for encryption.

IP(M) - This is applied on the plaintext M that is to be encrypted.
IP_INV(M) - This is applied after all 6 rounds of DES are done on message M.
S - There are 8 S-boxes. Each S-box has 6-bit input and a 4-bit output.

E(M) - Expand 32-bits of text M to 48-bits.
P(M) - This step permutes the 32-bit input M.
PC1 - Key permutation that maps 64 bits of the key to 56 bits and removes the parity bits
Shift - Shift that is performed on the key obtained as the output of PC1
PC2 - Key permutation that maps 56 bits of Shift's output to 48 bits

Methodology:

We perform differential cryptanalysis using two 3-round characteristics and used a chosen-plaintext attack for cryptanalysis of a 6-round DES. The characteristics used are 40080000 04000000 and 00200008 00000400.

Because one byte contains two characters, four bits are used to represent one character. We can only represent 16 characters with 4 bits, so we tried a few plaintexts and compared the ciphertexts to see which 16 characters are used in the game. We deduced from the ciphertexts that the alphabets f to u are used in the game. As a result, we began by mapping the letters f-u to the numbers 0-15.:

```
{
    'd' : '0000',
    'e' : '0001',
    'f' : '0010',
    'g' : '0011',
    'h' : '0100',
    'i' : '0101',
    'j' : '0110',
    'k' : '0111',
    'l' : '1000',
    'm' : '1001',
    'n' : '1010',
    'o' : '1011',
    'p' : '1100',
```

```
    'q' : '1101',
    'r' : '1110',
    's' : '1111'
}
```

The input and output size of one DES block is 64 bits i.e. 8 bytes (block size) which means 16 letters. Therefore, we decided to work on plaintexts of size 16 letters.

Step 1: Generation of Plaintext Pairs
The differential characteristic 40080000 04000000 with probability 1/16 and 00200008 00 000400 with probability 1/ 16 are used. We generated 1000 pairs of plaintexts and ciphertexts corresponding to each characteristic to break a 6-round DES. The first 2000 plaintext pairs are generated such that their XOR was 00 00 80 10 00 00 40 00, which is obtained by applying inverse initial permutation on the characteristic 40 08 00 00 04 00 00 00 and another 2000 plaintext pairs such that their XOR was 00 00 08 01 00 10 00 00, which is obtained by applying inverse initial permutation on the characteristic 00 20 00 08 00 00 04 00. These inputs are stored in **plaintexts1.txt** and **plaintexts2.txt**respectively. The code for the generation of plaintext pairs is in Random_plaintexts_generator.ipynb.

Step 2: Obtaining Ciphertexts corresponding to the Plaintexts

We used Python's pexpect to establish a connection to the server using valid credentials to automate the collection of ciphertexts corresponding to the plaintexts. The ciphertexts for the plaintexts stored in plaintexts1.txt were generated using server1.py, and the ciphertexts for the plaintexts stored in plaintexts2.txt were generated using Ciphertext2_generator.py. The **ciphertexts1.txt** and **ciphertexts2.txt** files contain these ciphertexts.

Step 3: Find the key bits of the K6 round key
Steps 3.1 to 3.4 were carried out for the ciphertexts obtained corresponding to each of the two characteristics.

3.1: We used the mapping of characters defined above to convert the obtained ciphertext to binary and then, we used CryptAnalysis.ipynb to apply to reverse final permutation on these binary ciphertexts to get (L_6R_6) and $(L'_6R'_6)$, which is the output of the 6^{th} round of DES. We know that, $R_5 = L_6$, therefore using the values R_5 and R'_5, we computed output of Expansion box and input XOR of S-boxes for 6^{th} round.

3.2: For the first characteristic mentioned above, $L_5 = 04000000$ and for the second characteristic $L_5 = 00000400$. We found output of permutation box by performing $L_5 \oplus (R_6 \oplus R'_6)$, then we applied inverse permutation on this value to obtain output XOR of S-boxes for 6^{th} round.

3.3: Let $E(R_5) = \alpha_1\alpha_2 \cdots \alpha_8$ and $E(R'_5) = \alpha'_1\alpha'_2 \cdots \alpha'_8$ and $\beta_i = \alpha_i \oplus k_{6,i}$ and $\beta'_i = \alpha'_i \oplus k_{6,i}$, where $|\alpha_i| = 6 = |\alpha'_i|$ and $k_6 = k_{6,1}k_{6,2} \cdots k_{6,8}$. At this point, we know $\alpha_i,\ \alpha'_i,\ \beta_i \oplus \beta'_i$ and $\gamma_i \oplus \gamma'_i$. We created a 8 * 64 key matrix to store the number of times a key $k \in [1, 64]$ satisfies the possibility of being a key to S_i box, where $i \in [1, 8]$.

3.4: We computed the set $X_i = (\beta, \beta')|\beta \oplus \beta' = \beta_i \oplus \beta'$ and $S(\beta) \oplus S(\beta') = \gamma_i \oplus \gamma'_i$.
Then, we found the key k, such that $\alpha_i \oplus k = \beta$ and $(\beta, \beta') \in X_i$ for some β'. For all the keys k which satisfied this condition for S_i box, we incremented their count in the key matrix i.e. key_matrix[i][k] was incremented.

- After performing the above analysis to find the keys, we obtained the following results for characteristic 40080000 04000000:

S-box	Max	Mean	Key	Diff
S1	157	67	45	90
S2	316	77	51	239
S3	116	68	37	48
S4	106	66	7	40
S5	148	66	28	82
S6	310	76	41	234
S7	185	74	13	111
S8	184	72	63	112

For this characteristic, in round 4, XOR will be zero for S2, S6, S8, S7 and S1. Therefore, in round 6 these S-boxes will give the corresponding key bits of K5. Also, it can be observed that a significant difference is seen in the maximum key frequency and mean key frequency for these S-boxes which further assures of these key values are correct. We proceeded by taking the key bits for S2, S6, S8, S7 and S1 boxes as 51, 41, 63, 13 and 45 respectively.

- The above analysis gave the following results for characteristic
00200008 00000400:

S-box	Max	Mean	Key	Diff
S1	149	66	45	83
S2	165	69	51	96
S3	116	65	37	51
S4	288	76	7	212
S5	165	66	28	99
S6	274	74	41	200
S7	114	64	13	50
S8	94	65	63	29

For this characteristic, in round 4, XOR will be zero for S4, S6, S5, S2 and S1. Therefore, in round 6 these S-boxes will give the corresponding key bits of K5. Also, it can be observed that a significant difference is seen in the maximum key frequency and mean key frequency for these S-boxes. We proceeded by taking the key bits for S4, S6, S5, S2 and S1 boxes as 7, 41, 28, 51 and 45 respectively.

Both the characteristics have S2, S1 and S6 as common S-boxes and we obtained the same key values for these three S-boxes which further verified that our computations so far are correct.

Therefore, we proceeded by taking key values for S1, S2, S4, S5, S6, S7 and S8 as 45, 51, 7, 28, 41, 13 and 63 for round key K5. Thus, at this point we know 42 bits of the 56-bit key.

Step 4: Find the Actual Key from 42 known bits
Next, we applied a key scheduling algorithm to obtain the actual positions of these known 42 bits in the 56-bit key and obtained the following result:

X11XX1XX01011X100XX11X11000X0101111X01111100X11X1001X001
(Master Key)
here X denotes unknown bits.

At this point, we have 14 unknown bits and for these 14 unknown bits of the DES key, we iterate through all 2^14 possible permutations of the key to find the correct key. We took plaintext= dddddddd dddddddd and the corresponding ciphertext= **kjpijsjqdslmeihr** and performed 6 round DES encryption. The key which encrypts this plaintext to produce the correct ciphertext is the final key. From this step, we obtained the following key which satisfied the above condition:

Actual 56 Bit Key=
01101110010111100111101100000101111001111100111110011001

0110110010111001110100000101110011110011110011001

After obtaining the 56-bit key, we found the 48 bit round key for each round.

ROUND KEY IN BINARY

Round 1 111011000100111100000111111111011111101010100011

Round 2 011011110011011101100010001110111110101100101101

Round 3 111010101101010011101101111100100101110110010110

Round 4 110110011100001101011010110011010010001110111111

Round 5 001001001101101110111011111101110111101011001001

Round 6 101101110011100101000111011100101001001101111111

No files uploaded

Q5 Password

5 Points

What was the password used to clear this level?

piqxkbgghq

Q6 Codes

0 Points

Unlike previous assignments, this time it is MANDATORY that you upload the codes used in the cryptanalysis. If you fail to do

so, you will be given 0 marks for the entire assignment.

Enigma_code_base.zip Download

```
1  Large file hidden. You can download it using the
   button above.
```

Assignment 4

GRADED

GROUP

Pranshu Sahijwani
Kajal Sethi
Gajender Sharma
View or edit group

TOTAL POINTS

67.5 / 100 pts

QUESTION 1

Team Name **0** / 0 pts

QUESTION 2

Commands **10** / 10 pts

QUESTION 3

CryptoSystem **5** / 5 pts

QUESTION 4

Analysis **70** / 80 pts

QUESTION 5

Password **5** / 5 pts

QUESTION 6

Codes **-22.5** / 0 pts