Network Security Assignment - 02
Project - Zero

Ketan Mohan Garg (2022248)
Keshav Bindlish (2022246)

Introduction

The Data Encryption Standard (DES) is a symmetric-key block cipher algorithm that encrypts and decrypts data in 64-bit blocks using a 56-bit key. This report provides an overview of the DES implementation in Python, including the working of each function and how the algorithm processes plaintext to produce ciphertext and vice versa.

CODE EXPLANATION

At starting tables used were initialised, that are ***initial_perm*** used for the initial permutation of the plaintext, ***final_perm*** used for the final permutation, ***Exp_box*** used for expanding 32 bits text to 48 bits text, ***S_Box*** 8 tables used for converting 6 bits binary to 4 bit binary ***perm_tab*** for permuting text at last stage of f-box, ***PC1*** and ***PC2*** are permuted choice tables to reduce the length of key.

Working of Each Function

1. **`hex_to_bin(hex_str)`**

   Converts a hexadecimal string to a 64-bit binary string.

2. **`bin_to_hex(bin_str)`**

   Converts a binary string to a hexadecimal string.

3. **`myxorfunc(bin_arr1, bin_arr2)`**

   Performs a bitwise XOR operation on two binary arrays.

4. **`initial_permutation(binary_input)`**

   Performs the initial permutation of the DES algorithm. Takes a 64-bit binary string as input. Uses the `initial_perm` table to rearrange the bits.

5. **`final_permutation(binary_input)`**

   Performs the final permutation of the DES algorithm. Takes a 64-bit binary string as input. Uses the `final_perm` table to rearrange the bits.

6. **`expansionbox(R)`**

   Expands the 32-bit right half of the data to 48 bits using the expansion table.

7. **`s_box(input, s_box)`**

   Performs substitution using the S-boxes. Takes a 6-bit binary string and an S-box as input. Determines the row and column indices from the input. Look up the S-box to find the 4-bit output. Returns the 4-bit binary string.

8. **`myffunc(Rp, subkey)`**

   Implements the DES round function (F-function). Expands the 32-bit right half to 48 bits using `expansionbox()`. XORs the expanded right half with the subkey using `myxorfunc()`. Splits the result into 8 groups of 6 bits and applies the S-boxes. Permutes the S-box output using the `perm_tab` table.

9. **`left_shift(bits, shift_count)`**

   Performs a left circular shift on a binary string.

10. **`myroundkeyfunc(key_bin)`**

    Generates the 16 subkeys for the DES algorithm. Applies the `PC1` permutation to the 64-bit key to produce a 56-bit key. Splits the key into two 28-bit halves. Performs left shifts on both halves according to the `shift_table`. Combines the halves and applies the `PC2` permutation to produce a 48-bit subkey. Repeats the process for all 16 rounds. Returns the list of subkeys.

11. **`des_encrypt(plaintext, key)`**

    Encrypts the plaintext using DES. Converts the plaintext and key to binary. Performs the initial permutation. Splits the data into left and right halves. Applies the F-function and XORs the result with the left half for 16 rounds. Swaps the left and right halves after each round. Performs the final permutation and returns the ciphertext.

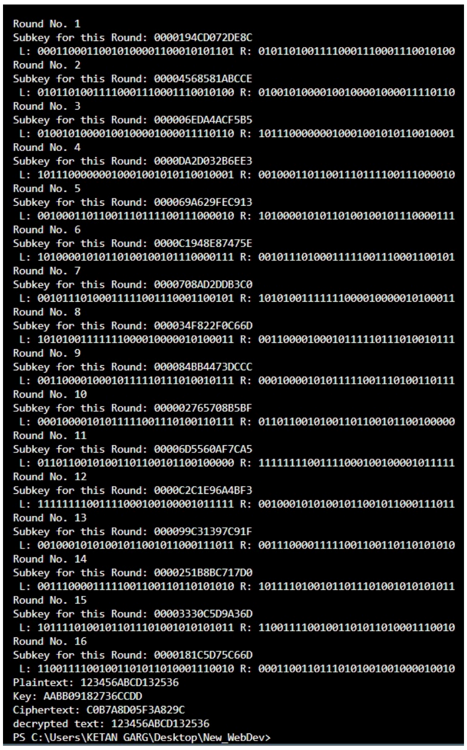
12. **`des_decrypt(ciphertext, key)`**

    Decrypts the ciphertext using DES. Converts the ciphertext and key to binary. Performs the initial permutation. Splits the data into left and right halves. Applies the F-function and XORs the result with the left half for 16 rounds (using subkeys in reverse order). Swaps the left and right halves after each round. Performs the final permutation and returns the decrypted plaintext.

13. **`main(plaintext, key)`**

Demonstrates the encryption and decryption process. Calls `des_encrypt()` and `des_decrypt()`. Prints the plaintext, key, ciphertext, and decrypted text.

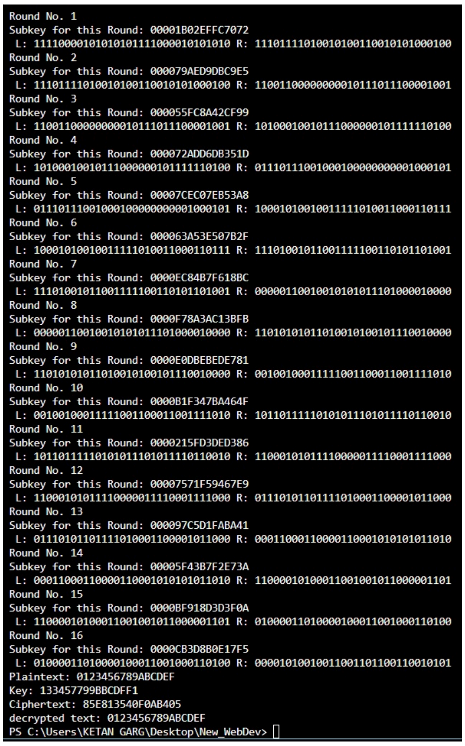
Output:



```
Round No. 1
Subkey for this Round: 0000194CD072DE8C
 L: 00011000110010100001100010101101 R: 01011010011110001110001110010100
Round No. 2
Subkey for this Round: 00004568581ABCCE
 L: 01011010011110001110001110010100 R: 01001010000100100001000011110110
Round No. 3
Subkey for this Round: 000006EDA4ACF5B5
 L: 01001010000100100001000011110110 R: 10111000000010001001010110010001
Round No. 4
Subkey for this Round: 0000DA2D032B6EE3
 L: 10111000000010001001010110010001 R: 00100011011001110111100111000010
Round No. 5
Subkey for this Round: 000069A629FEC913
 L: 00100011011001110111100111000010 R: 10100001010110100100101110000111
Round No. 6
Subkey for this Round: 0000C1948E87475E
 L: 10100001010110100100101110000111 R: 00101110100011111001110001100101
Round No. 7
Subkey for this Round: 0000708AD2DDB3C0
 L: 00101110100011111001110001100101 R: 10101001111111000010000010100011
Round No. 8
Subkey for this Round: 000034F822F0C66D
 L: 10101001111111000010000010100011 R: 00110000100010111110111010010111
Round No. 9
Subkey for this Round: 000084BB4473DCCC
 L: 00110000100010111110111010010111 R: 00010000101011111001110100110111
Round No. 10
Subkey for this Round: 000002765708B5BF
 L: 00010000101011111001110100110111 R: 01101100101001101100101100100000
Round No. 11
Subkey for this Round: 00006D5560AF7CA5
 L: 01101100101001101100101100100000 R: 11111110011110001001000010111111
Round No. 12
Subkey for this Round: 0000C2C1E96A4BF3
 L: 11111110011110001001000010111111 R: 00100010101001011001011000111011
Round No. 13
Subkey for this Round: 000099C31397C91F
 L: 00100010101001011001011000111011 R: 00111000011111001100110110101010
Round No. 14
Subkey for this Round: 0000251B8BC717D0
 L: 00111000011111001100110110101010 R: 10111101001011011101001010101011
Round No. 15
Subkey for this Round: 00003330C5D9A36D
 L: 10111101001011011101001010101011 R: 11001111001001101011010001110010
Round No. 16
Subkey for this Round: 0000181C5D75C66D
 L: 11001111001001101011010001110010 R: 00011001101110101001001000010010
Plaintext: 123456ABCD132536
Key: AABB09182736CCDD
Ciphertext: C0B7A8D05F3A829C
decrypted text: 123456ABCD132536
PS C:\Users\KETAN GARG\Desktop\New_WebDev>
```

Plaintext = "123456ABCD132536"

Key = "AABB09182736CCDD"



```
Round No. 1
Subkey for this Round: 00001B02EFFC7072
 L: 11110000101010101111000010101010 R: 11101111010010100110010101000100
Round No. 2
Subkey for this Round: 000079AED9DBC9E5
 L: 11101111010010100110010101000100 R: 11001100000000010111011100001001
Round No. 3
Subkey for this Round: 000055FC8A42CF99
 L: 11001100000000010111011100001001 R: 10100010010111000000101111110100
Round No. 4
Subkey for this Round: 000072ADD6DB351D
 L: 10100010010111000000101111110100 R: 01110111001000100000000001000101
Round No. 5
Subkey for this Round: 00007CEC07EB53A8
 L: 01110111001000100000000001000101 R: 10001010010011111010011000110111
Round No. 6
Subkey for this Round: 000063A53E507B2F
 L: 10001010010011111010011000110111 R: 11101001011001111100110101101001
Round No. 7
Subkey for this Round: 0000EC84B7F618BC
 L: 11101001011001111100110101101001 R: 00000110010010101011101000010000
Round No. 8
Subkey for this Round: 0000F78A3AC13BFB
 L: 00000110010010101011101000010000 R: 11010101011010010100101110010000
Round No. 9
Subkey for this Round: 0000E0DBEBEDE781
 L: 11010101011010010100101110010000 R: 00100100011111001100011001111010
Round No. 10
Subkey for this Round: 0000B1F347BA464F
 L: 00100100011111001100011001111010 R: 10110111110101011101011110110010
Round No. 11
Subkey for this Round: 0000215FD3DED386
 L: 10110111110101011101011110110010 R: 11000101011110000011110001111000
Round No. 12
Subkey for this Round: 00007571F59467E9
 L: 11000101011110000011110001111000 R: 01110101101111010001100001011000
Round No. 13
Subkey for this Round: 000097C5D1FABA41
 L: 01110101101111010001100001011000 R: 00011000110000110001010101011010
Round No. 14
Subkey for this Round: 00005F43B7F2E73A
 L: 00011000110000110001010101011010 R: 11000010100011001001011000001101
Round No. 15
Subkey for this Round: 0000BF918D3D3F0A
 L: 11000010100011001001011000001101 R: 01000011010000100011001000110100
Round No. 16
Subkey for this Round: 0000CB3D8B0E17F5
 L: 01000011010000100011001000110100 R: 00001010010011001101100110010101
Plaintext: 0123456789ABCDEF
Key: 133457799BBCDFF1
Ciphertext: 85E813540F0AB405
decrypted text: 0123456789ABCDEF
PS C:\Users\KETAN GARG\Desktop\New_WebDev>
```

Plaintext = "0123456789ABCDEF"

Key = "133457799BBCDFF1"