Ministry of Education, Research and Culture
Technical University of Moldova
Software Engineering and Automation Departments

REPORT

Laboratory work No. 4

Course: Cryptography and Security

Theme: Block Cipher. DES

Author: Corneliu Catlabuga
FAF-213

Checked by: univ. assist.
Cătălin Mîțu

Chisinau 2023

Objective:

Understand and implement an element of the DES algorithm.

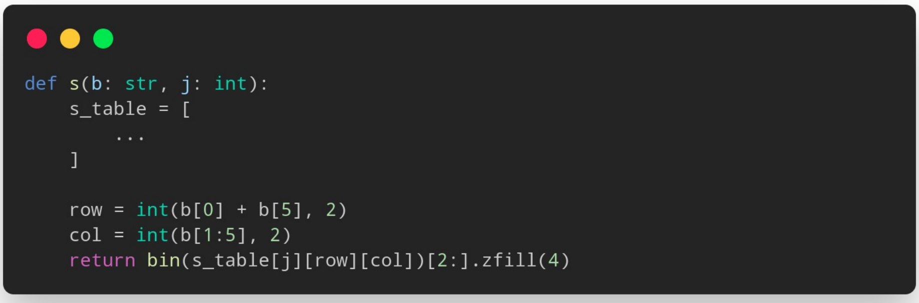
Task:

Get Sj(Bj) for a given 48-bit block from i-th round of the DES algorithm for the XOR permutation K_i+E_i(B_{i-1}).

Theoretical considerations:

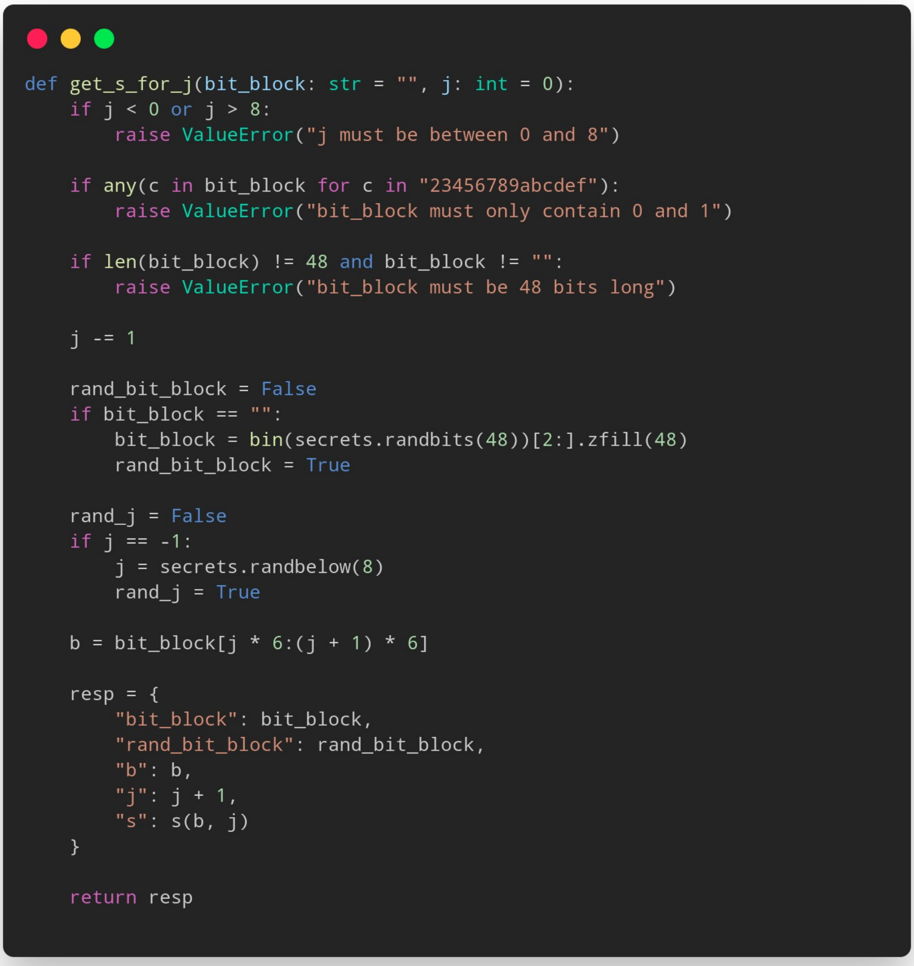
The Data Encryption Standard (DES) is a symmetric-key block cipher algorithm used for encrypting and decrypting data. It operates on 64-bit blocks of data and uses a 56-bit key to control the encryption process. DES employs a series of 16 rounds of substitution, permutation, and XOR operations to transform the plaintext into ciphertext.

Implementation, practical results:



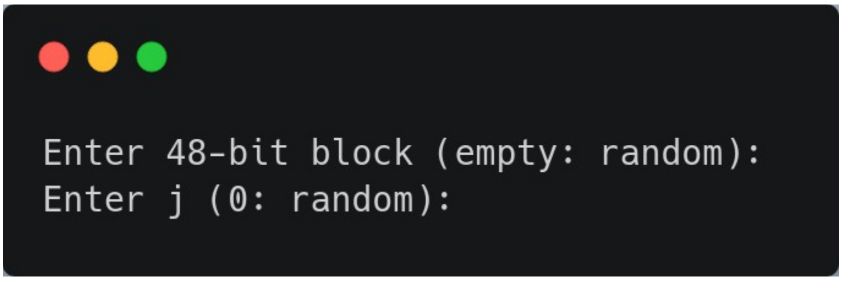
```python
def s(b: str, j: int):
    s_table = [
        ...
    ]

    row = int(b[0] + b[5], 2)
    col = int(b[1:5], 2)
    return bin(s_table[j][row][col])[2:].zfill(4)
```

Figure 1: $S_j(B_{j-1})$ function



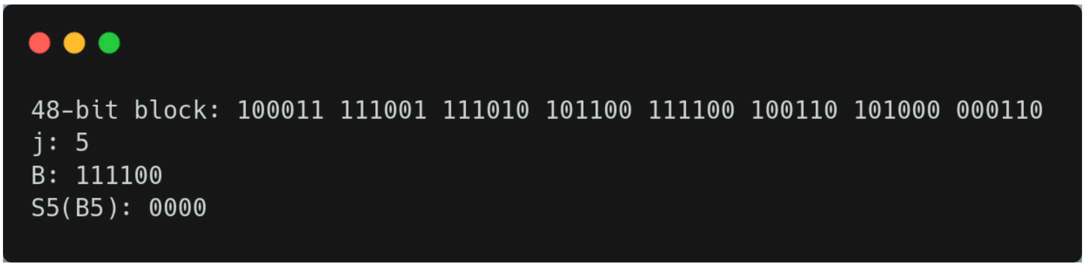
```python
def get_s_for_j(bit_block: str = "", j: int = 0):
    if j < 0 or j > 8:
        raise ValueError("j must be between 0 and 8")

    if any(c in bit_block for c in "23456789abcdef"):
        raise ValueError("bit_block must only contain 0 and 1")

    if len(bit_block) != 48 and bit_block != "":
        raise ValueError("bit_block must be 48 bits long")

    j -= 1

    rand_bit_block = False
    if bit_block == "":
        bit_block = bin(secrets.randbits(48))[2:].zfill(48)
        rand_bit_block = True

    rand_j = False
    if j == -1:
        j = secrets.randbelow(8)
        rand_j = True

    b = bit_block[j * 6:(j + 1) * 6]

    resp = {
        "bit_block": bit_block,
        "rand_bit_block": rand_bit_block,
        "b": b,
        "j": j + 1,
        "s": s(b, j)
    }

    return resp
```

Figure 2: Final function for $S_j(B_{j-1})$



```
Enter 48-bit block (empty: random):
Enter j (0: random):
```

Figure 3: Input prompt



```
48-bit block: 100011 111001 111010 101100 111100 100110 101000 000110
j: 5
B: 111100
S5(B5): 0000
```

Figure 4: Output

Conclusions:

In the Data Encryption Standard (DES) algorithm, computing $S_j(B_j)$ for the output of $K_i+E(B_{i-1})$ plays a crucial role in the substitution (S-box) layer of the encryption process. This step involves replacing blocks of bits with different values based on predefined substitution tables, which adds a non-linear and confounding element to the algorithm.

References:

1. Github repo: https://github.com/muffindud/CS_Lab/tree/lab4/lab4