Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИУ «Информатика и системы управления»

КАФЕДРА ИУ-7 «Программное обеспечение ЭВМ и информационные технологии»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №2

по дисциплине «Защита информации»

«Алгоритм шифрования DES»

Студент группы **ИУ7-72Б** _____ **Фам М. Х.**

Руководитель _____ **Чиж И. С.**

2024 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Шифрование информации — занятие, которым человек занимался ещё до начала первого тысячелетия, занятие, позволяющее защитить информацию от посторонних лиц.

Шифровальная алгоритм DES — алгоритм, разработанный в 1977 году компанией IBM и являющийся официальным стандартом шифрования.

Целью данной работы является реализация в виде программы на языке программирования C или C++ шифровального алгоритма DES в режиме работы CBC.

Для достижения поставленной цели необходимо выполнить следующие задачи:

1) изучить шифроовальный алгоритм DES и его режим работы CBC;
2) реализовать шифровальный алгоритм DES в виде программы, обеспечив возможности шифрования и расшифровки файла в режиме работы CBC;
3) описать и обосновать полученные результаты в отчёте о выполненной лабораторной работе.

1 Аналитическая часть

1.1 Алгоритм DES

Шифровальная алгоритм DES (англ. *Data Encryption Standart* — DES) — симметричный шифровальный алгоритм, разработанный в 1977 году компанией IBM. Он использует блочное шифрование, длина блока фиксирована и равна 64 битам. Однако каждые 8 бит в ключе игнорируются, что приводит к правильной длине ключа 56 бит в DES. Однако в любом случае один блок на 64 бита является вечной организацией DES. Он состоит из 3 следующих шагов, рисунок 1.1:

— начальная перестановка (англ. *Initial Permutation* — IP), во время которой биты переставляются в порядке, определённом в специальной таблице;

— 16 раундов шифрования;

— завершающей перестановки (англ. *Final Permutation* – FP), соовершающей преобразования, обратные сделанным на первом шаге.

Раунд шифрования состоит из 5 следующих этапов

1) расширение (англ. *expansion* — E);
2) получение ключа раунда (англ. *Round Key* – RK);
3) скремблирование (англ. *substitution* — S);
4) перестановка (англ. *permutation* — P)
5) смешивание ключа (англ. *key mixing* — KM).

Рисунок 1.1 – Обобщенная схема шифрования в алгоритме DES

Расширение, во время которого каждая из половин блока шифрования по 32 бит дополняется путём перестановки и дублировоания бит до длины в 48 бит.

Получение ключа раунда необходимо для применения в раунде шифрования 48-битного ключа раунда, полученного из основного ключа DES. Основной ключ имеет длину 64 бита, однако значащих бит из 64 всего 56, остальные добавлены для избыточности и контроля передачи ключа. Из этих 56 бит получают 48 путём разбиения на равные части и применению битовой операции циклического сдвига и нахождению нового значения посредством специальной таблицы.

Скремблирование предназначено для получения из 48-битного потока 32-битного путём разбиения на 6 частей по 8 бит и обработки каждой части в S-блоках (англ. *Substitution boxes*), которые заменяют блоки с длиной 6 бит на блоки 4 бит посредством использования специальной таблицы.

Перестановка представляет из себя перемешивания полученной после-

довательности из 32 бит при помощи таблицы перемешивания.

Смешивание ключа представляет из себя операцию XOR полученного 32-битного значения с ключом раунда.

1.2 DES-CBC

На рисунке 1.2 приведена схема алгоритма DES в режиме CBC.

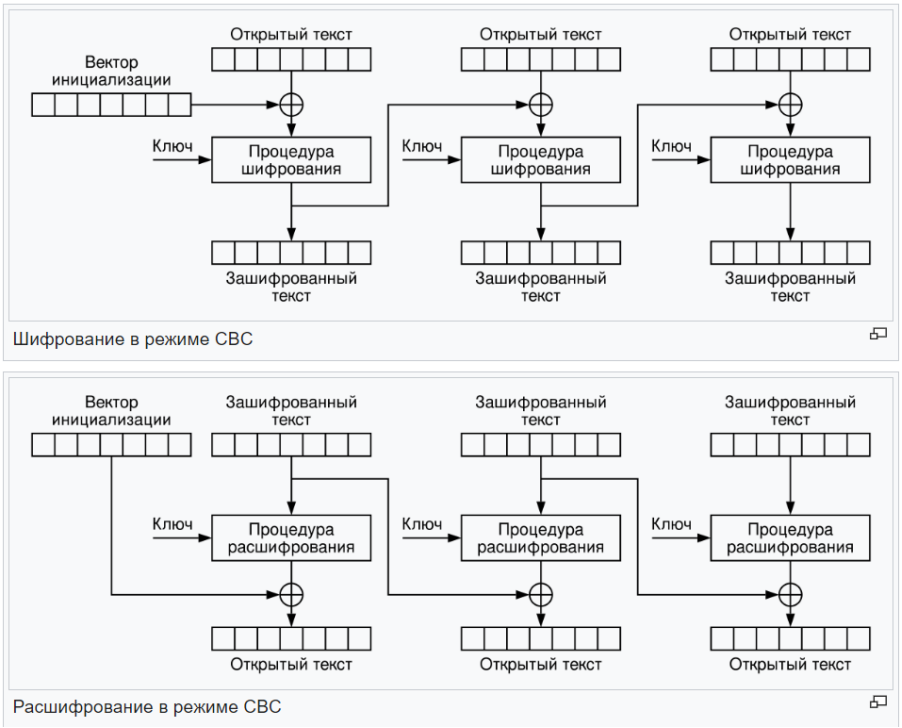

Рисунок 1.2 – Режим сцепления блоков шифротекста CBC

2 Конструкторская часть

В этом разделе представлена схема алгоритма шифровальной машины «Энигма».

2.1 Разработка алгоритмов

На рисунках ??–2.3 представлены схемы алгоритмов DES, раунда DES, функции Фейстеля.

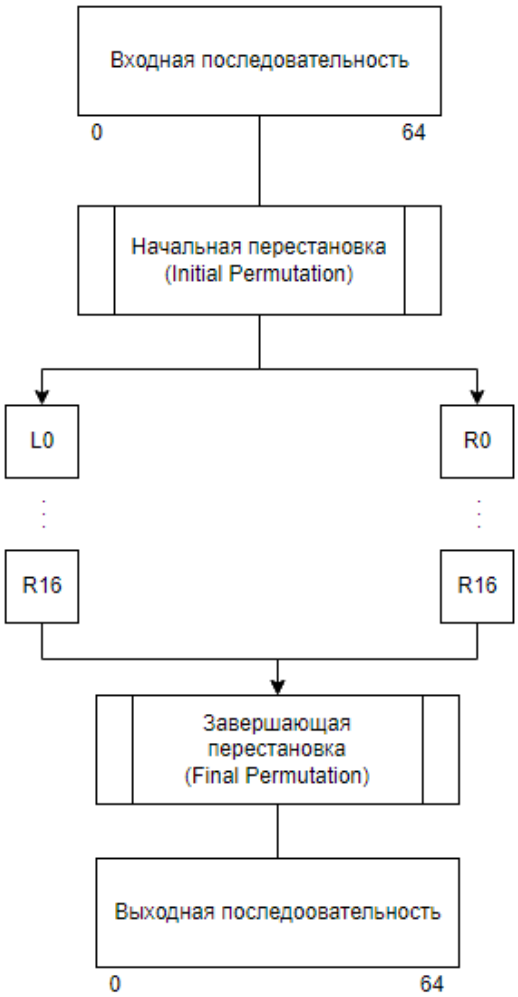

Рисунок 2.1 – Схема алгоритма DES

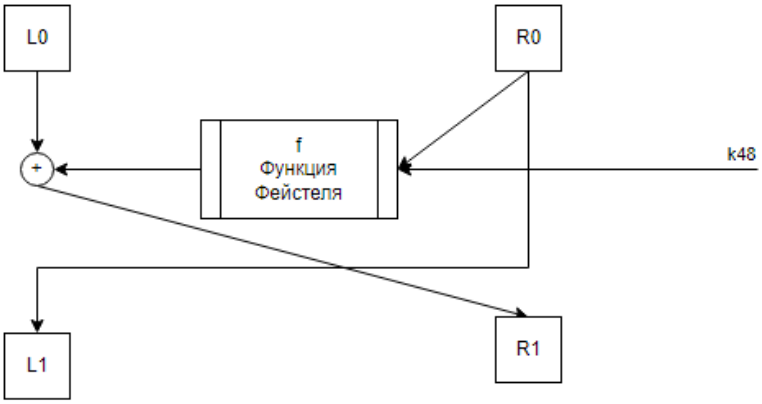

Рисунок 2.2 – Схема алгоритма раунда DES

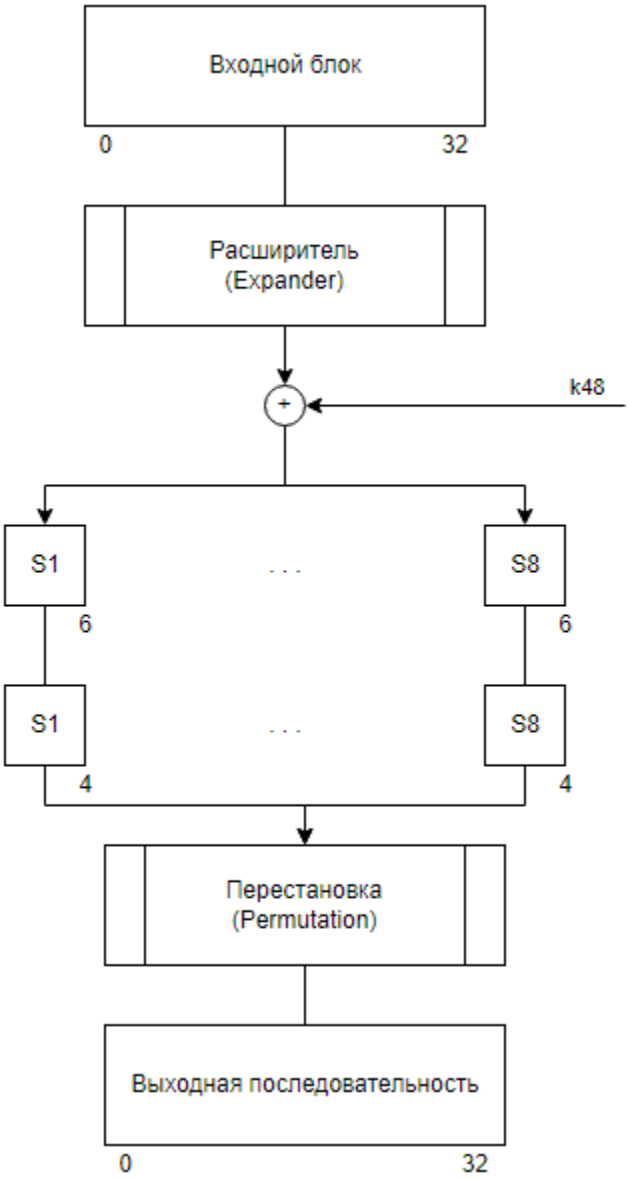

Рисунок 2.3 – Схема алгоритма функции Фейстеля

3 Технологическая часть

3.1 Средства реализации

Для программной реализации шифровальной машины был выбран язык C++ [1]. В данном языке есть все требующиеся инструменты для данной лабораторной работы. В качестве среды разработки была выбрана среда Visual Studio 2022 [2].

3.2 Реализация алгоритма

Листинг 3.1 – Реализация алгоритма DES

```
1    bitset<64> process_block(bitset<64> value, bitset<64> key,
        bool decrypte = false)
2    {
3        // generate keys
4        auto keys = generate_keys(key, decrypte);
5
6        // initial permutation
7        auto round_val = IP_f(value);
8
9        // 16 rounds
10       for (auto rkey : keys) {
11           round_val = wround(round_val, rkey);
12       }
13
14       // final permutation
15       auto final_val = FP_f(round_val);
16
17       return final_val;
18   }
```

Листинг 3.2 – Реализация метода шифрования и дешифрования DES в режиме CBC

```
vector<char> cypher(vector<char> input, vector<char> key,
    vector<char> vi)
{
    vector<char> buffer = {};
    vector<char> result = {};

    int last_cnt = 0;

    auto vi_initial = vchar_to_bitset64(vi);
    auto key_b = vchar_to_bitset64(key);


    for (auto sym : input) {
        if (buffer.size() < 8) {
            buffer.push_back(sym);
        }

        if (buffer.size() == 8) {
            // buffer size is 8 -> can be cyphered
            auto buf_b = vchar_to_bitset64(buffer) ^ vi_initial;

            auto tmp_b_1 = _des.process_block(buf_b, key_b);

            vi_initial = tmp_b_1;

            auto tmp_res = bitset64_to_vchar(tmp_b_1);

            result.insert(result.end(), tmp_res.begin(),
                tmp_res.end());

            buffer.clear();
        }
```

```
31      }
32
33      if (buffer.size() > 0 && buffer.size() < 8) {
34          while (buffer.size() < 8) {
35              buffer.push_back((char)0);
36              last_cnt += 1;
37          }
38          auto buf_b = vchar_to_bitset64(buffer) ^ vi_initial;
39
40          auto tmp_b_1 = _des.process_block(buf_b, key_b);
41
42          vi_initial = tmp_b_1;
43
44          auto tmp_res = bitset64_to_vchar(tmp_b_1);
45
46          result.insert(result.end(), tmp_res.begin(),
                tmp_res.end());
47      }
48
49      result.push_back((char)last_cnt);
50
51
52      return result;
53  }
54  }
```

Вывод

В данном разделе были рассмотрены средства реализации, а также представлены листинги реализации шифровального алгоритма DES и режима работы CBC.

Заключение

В результате лабораторной работы был реализован в виде программы шифровальный алгоритма DES в режиме работы CBC.

Был и выполнены следующие задачи:

1) изучен шифроовальный алгоритм DES и его режим работы CBC;
2) реализован шифровальный алгоритм DES в виде программы, обеспечена возможность шифрования и расшифровки файла в режиме работы CBC;
3) описаны и обоснованы полученные результаты в отчёте о выполненной лабораторной работе.

Список использованных источников

1. Язык программирования C++. https://learn.microsoft.com/en-us/cpp/cpp/cpp-language-reference?view=msvc-170. дата обращения: 15.10.2024.

2. Visual Studio 2022. https://visualstudio.microsoft.com/vs/. дата обращения: 15.10.2024.