МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

**Отчет по лабораторной работе №7**

**″Исследование блочных шифров″**

Выполнила студентка 3 курса 5 группы Максимчикова Ю. С.

Проверила: Блинова Е. А.

Минск 2020

**Цель**: изучение и приобретение практических навыков разработки и использования приложений для реализации блочных шифров.

**Задачи**:

* Закрепить теоретические знания по алгебраическому описанию, алгоритмам реализации операций зашифрования/расшифрования и оценке криптостойкости блочных шифров.
* Разработать приложение для реализации указанных преподавателем методов блочного зашифрования/расшифрования.
* Выполнить анализ криптостойкости блочных шифров.
* Оценить скорость зашифрования/расшифрования реализованных шифров.
* Оценить скорость зашифрования/расшифрования реализованных способов шифров.
* Результаты выполнения лабораторной работы оформить в виде описания разработанного приложения, методики выполнения экспериментов с использованием приложения и результатов эксперимента.

**Теория.** Блочное зашифрование (расшифрование) предполагает разбиение исходного открытого (зашифрованного) текста на равные блоки, к которым применяется однотипная процедура зашифрования (расшифрования). Указанная однотипность характеризуется, прежде всего, тем, что процедура зашифрования (расшифрования) состоит из совокупности повторяющихся наборов преобразований, называемых раундами.

Основные требования к шифрам рассматриваемого класса можно сформулировать следующим образом:

* даже незначительное изменение исходного сообщения должно приводить к существенному изменению зашифрованного сообщения;
* устойчивость к атакам по выбранному тексту;
* алгоритмы зашифрования/расшифрования должны быть реализуемыми на различных платформах;
* алгоритмы должны базироваться на простых операциях;
* алгоритмы должны быть простыми для написания кода, вероятность появления программных ошибок должна быть низкой;
* алгоритмы должны допускать их модификацию при переходе на иные требования по уровню криптостойкости.

В данной лабораторной работе необходимо было разработать авторское приложение в соответствии с целью лабораторной работы. Для варианта номер 11 в качестве рассматриваемого метода был взят алгоритм DES c ключом, состоящим из первых восьми символов собственных фамилии и имени.

Итак, для того, чтобы зашифровать сообщение алгоритмом DES, необходимо выполнить следующую последовательность шагов:

* довести исходное сообщение до такого размера (в битах), чтобы оно нацело делилось на размер блока;
* разделить исходное сообщение на блоки;
* довести длину ключа до длины половины блока;
* перевести ключ в бинарный формат (в нули и единицы);
* провести над каждым блоком прямое преобразование сетью Фейстеля в течении 16-ти раундов. После каждого раунда необходимо выполнять циклический сдвиг ключа на заданное количество символов;
* соединить все блоки вместе; таким образом получим сообщение, зашифрованное алгоритмом DES.

Расшифровка DES производится по аналогии. Используется обратное преобразование сетью Фейстеля.

Программный код разработанного в данной лабораторной работе программного средства представлен на рисунках. Алгоритм DES реализован без использования каких-либо стандартных библиотек. Рассмотрим его более детально.

На рисунке 1 представлен код функции, выполняющей преобразование исходного текста в его бинарное представление.

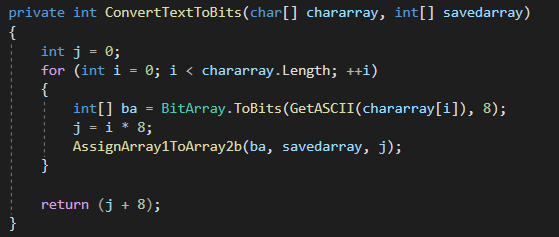


Рис. 1 – Функция преобразования исходного текста в его бинарный вид

На рисунке 2 представлен код функции, выполняющей доведение бинарного представления исходного текста до размера, который делится на размер блока, а именно 64 бит. Это происходит путем добавления в конец текста нулей.

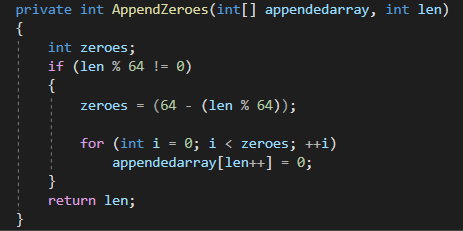


Рис. 2 – Функция доведения размера блока

Функции преобразования в бинарный вид и доведения размера выполняются и для ключа. Далее над ключом выполняется функция, отбрасывающая каждый восьмой бит ключа, чтобы привести ключ к размеру в 56 бит. Код этой функции представлен на рисунке 3.

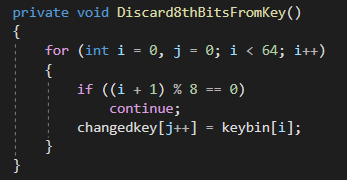


Рис. 3 – Функция, отбрасывающая каждый восьмой бит ключа

Далее в цикле разбиваем текст на блоки длиной в 64 бита и выполняем следующий алгоритм над каждый таким блоком. В начале над блоком выполняется начальная перестановка в соответствии с таблицей, представленной на рисунке 4.

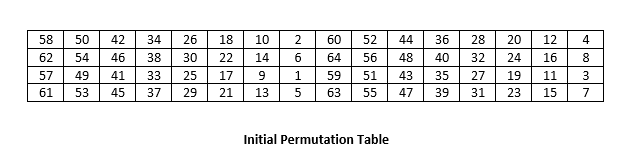


Рис. 4 – Таблица для начальной перестановки

В соответствии с этой таблицей на месте первого бита преобразованного блока становится 58ый бит исходного блока, на месте второго – 50ый и т.д. Функция, выполняющая начальную перестановку представлена на рисунке 5.

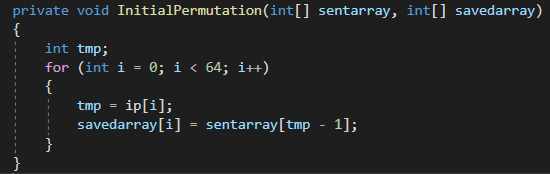


Рис. 5 – Функция начальной перестановки

После начальной перестановки над блоком выполняется функция, которая делит его на левую и правую части, где каждая размером в 32 бита. Код этой функции представлен на рисунке 6.

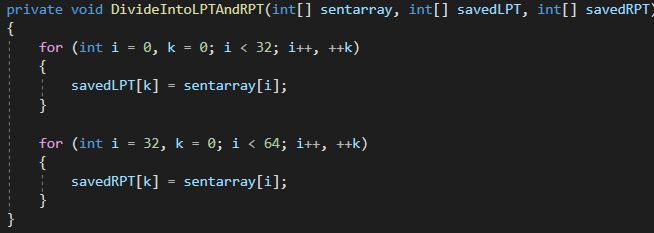


Рис. 6 – Функция, разбивающая блок на левую и правую часть

Далее над ключом выполняется циклический сдвиг. Функция на рисунке 7.

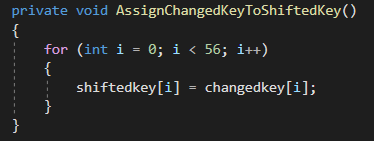


Рис. 7 – Функция, выполняющая циклический сдвиг над ключом

Следующий этап в алгоритме DES подразумевает выполнение одних и тех же действий в 16 раундов. Рассмотрим, что происходит за время оного раунда. Сохраняем значение правой части блока во временной переменной. Делим ключ на две части по 28 бит. Выполняем циклические сдвиги над каждой из частей. Соединяем полученные части и получаем ключ длиной в 56 бит. Далее выполняем сжатие ключа до 48 бит, одновременно выполняя перестановку в соответствии с таблицей. Далее расширяем правую часть блока до 48 бит. Далее выполняем исключающее ИЛИ над расширенным правой частью блока и сжатым ключом. Результат данной операции теперь и является правой частью блока. Затем полученное 48-битное значение подается на вход блока подстановки S (от англ. Substitution – подстановка), результатом которой является 32-битное значение. Подстановка выполняется в восьми блоках подстановки или восьми S-блоках (S-boxes). При выполнении этой операции 48 битов данных делятся на восемь 6-битных подблоков, каждый из которых по соответствующей таблице замен замещается четырьмя битами. Подстановка с помощью S-блоков является одним из важнейших этапом DES. Таблицы замен для этой операции специально спроектированы так, чтобы обеспечивать максимальную криптостойкость. В результате выполнения этого этапа получаются восемь 4-битных блоков, которые вновь объединяются в единое 32-битное значение. Далее полученное 32-битное значение обрабатывается с помощью перестановки Р (от англ. Permutation – перестановка), которая не зависит от используемого ключа. Целью перестановки является максимальное переупорядочивание битов такое, чтобы в следующем раунде шифрования каждый бит с большой вероятностью обрабатывался другим S-блоком. И, наконец, результат перестановки объединяется с помощью операции XOR с левой половиной первоначального 64-битового блока данных. Затем левая и правая половины меняются местами, и начинается следующий раунд. После выполнения 16 раундов левая и правая части блока соединяются и выполняется финальная перестановка. Вышеописанные действия производятся над каждым 64-битным блоком. Финальным этапом шифрования является конкатенация полученных блоков в сплошной текст и преобразование бинарного представления текста в символьное. Результат работы алгоритма и замеры времени выполнения операция шифрования и расшифрования слова cryptography представлены на рисунке 8.

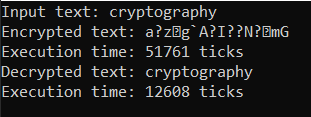


Рис. 8 – Результат работы приложения

Вывод: в результате данной лабораторной работы было разработано приложение для выполнения зашифрования и дешифрования текстов с помощью блочного шифра под названием DES. Также был проведен анализ его криптостойкости.