МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «КубГУ»)

Кафедра информационных технологий

КУРСОВАЯ РАБОТА

РАЗРАБОТКА МАТЕМАТИЧЕСКОЙ МОДЕЛИ КРИПТОСИСТЕМЫ НА ОСНОВЕ ОБОБЩЕННОГО РЮКЗАКА С ПРИМЕНЕНИЕМ КОДОВ ВАРШАМОВА

Работу выполнил _____ Д.А.Финджоян
(подпись, дата) (инициалы, фамилия)

Факультет компьютерных технологий и прикладной математики 3 курс

Направление 01.03.02 – «Прикладная математика и информатика»

Научный руководитель, проф.
д-р. физ.-мат. наук, доц. _____ В.О.Осипян
(подпись, дата) (инициалы, фамилия)

Нормоконтролер, ст. преп. _____А.В.Харченко
(подпись, дата) (инициалы, фамилия)

Краснодар 2017

РЕФЕРАТ

Курсовая работа 46 с., 11 ч., 18 рис., 1 табл., 15 источников.

КРИПТОГРАФИЯ, АСИММЕТРИЧНОЕ ШИФРОВАНИЕ, СИММЕТРИЧНОЕ ШИФРОВАНИЕ, КРИПТОГРАФИЧЕСКАЯ УСТОЙЧИВОСТЬ, ЗАДАЧА О РЮКЗАКЕ, АЛГОРИТМ МЕРКЛА – ХЕЛЛМАНА, ШИФРОВАНИЕ

Объектом исследования является криптографическая система защиты информации на основе двоичного рюкзака.

Цель работы – ознакомление с основами криптографии, разработка системы защиты информации на основе двоичного рюкзака.

В результате исследования были рассмотрены основные понятия в криптографии, а также разработана система защиты информации, основанная на алгоритме Меркла – Хеллмана.

Разработанная система защиты информации позволяет шифровать/расшифровывать файлы данных, а также генерировать открытые и секретные ключи. Время работы программы определяется объемом входных данных. Программа имеет простой интерфейс, а для работы использует «алфавит», который состоит из заглавных букв английского алфавита, пробела, цифр 0 – 9, запятой, точки, восклицательного знака, вопросительного знака, а также служебного символа «\n». Каждый символ можно представить 6 – ю битами информации. При использовании правильно выбранной пары ключей шифрование/расшифровывание данных производится однозначно (это означает, что ключ, которым можно зашифровать информацию, должен быть получен сильным модульным умножением относительно секретного ключа; для расшифровки необходимо использовать соответствующий секретный ключ).

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Основные методы повышения надежности цифровых устройств систем связи сводятся к разработке новых методов введения избыточности при кодировании информации. Это позволяет создавать устройства, которые с высоким шансом обнаруживают ошибки при передаче информации, причем ошибки также исправляются аппаратно и программно. Однако, с другой стороны нужно разрабатывать методы, которые могли бы предупреждать, предотвращать искажения информации, уничтожение или её несанкционированное использование.

Цель данной курсовой работы – создание математической модели криптосистемы на основе обобщенного рюкзака с применением кодов Варшамова. Будут даны математические обозначения ключевых элементов в такой криптосистеме, также будут упомянуты другие виды рюкзачных криптосистем, дано определение кодов Варшамова и пояснено их использование в криптосистеме. Наконец, будет в подробностях рассмотрена реализация криптосистемы и её функционал.

1 Рюкзачные криптосистемы

Основой рюкзачных криптосистем является задача о рюкзаке.

Постановка задачи о рюкзаке[1, с.160]:

Задан набор A из n различных положительных целых чисел $a_1, \ldots, a_n$, где a_i – вес определенной вещи в рюкзаке. Также задано некоторое число v – общая вместимость рюкзака. Каждая вещь a_i имеет ценность $c_i, i \in [1, n]$. Требуется найти такое подмножество A, чтобы в сумме элементы этого подмножества давали ровно k, а суммарная ценность была наибольшей.

Математически задачу можно сформулировать следующим образом[2]: имеется n грузов. Для каждого груза i определен его вес $a_i > 0$ и ценность $c_i > 0$. Ограничение суммарного веса задается параметром W. Необходимо максимизировать

$$\sum_{i=1}^{n} c_i x_i \tag{1}$$

с ограничениями

$$\sum_{i=1}^{n} a_i x_i \leq W \text{ и } x_i \in \{0,1\}. \tag{2}$$

Существует несколько вариантов задачи о ранце:

– Рюкзак 0-1(англ. 0-1 Knapsack Problem)[2, с.2]: каждый предмет не может входить в рюкзак более одного раза;

– Ограниченный рюкзак(англ. Bounded Knapsack Problem)[3, с.127]: каждый предмет не может входить в рюкзак более заданного числа раз;

– Неограниченный рюкзак(англ. Unbounded Knapsack Problem)[3, с.127]: каждый предмет может входить в рюкзак сколь угодно много раз;

– Рюкзак с мультивыбором(англ. Multiple-choice Knapsack Problem)[3, с.147]: все предметы делят на группы, и из каждой группы нужно выбрать лишь один предмет;

– Множественный рюкзак(англ. Multiple Knapsack Problem)[2, с. 157]: дано несколько рюкзаков, каждый из них имеет свою максимальную вместимость. Любой предмет можно положить в любой рюкзак или оставить;

– Многомерный рюкзак(англ. Multi-dimensional Knapsack Problem)[3, с.147]: вместо одного параметра(вес) у каждого предмета есть множество параметров (вес, объем, ценность и т.д.). Требуется найти такое подмножество предметов, чтобы затраты ресурсов не превышали их максимума, но ценность была максимальной;

– Квадратичная задача о рюкзаке(англ. Quadratic Knapsack Problem)[4, с.132]: общая ценность задается неотрицательно определенной квадратичной формой;

– Универсальный рюкзак[5]: коэффициенты повторов компонентов рюкзака берутся различными между собой способами из двух различных целочисленных массивов.

– Нестандартный рюкзак[6]: максимальное количество повторов вхождения каждого предмета не ограничено каким то одним числом, но может быть любым для конкретного предмета.

Одной из таких криптосистем является криптосистема Меркла – Хеллмана[7]. Её основа – стандартная задача о рюкзаке: предметы либо входят в рюкзак, либо нет. Если набор предметов публично известен, то сообщение можно представить как набор передаваемых предметов, а отправлять по каналу связи суммарный вес. Для шифрования текста нужно разбить его на блоки, длина каждого блока – n бит. Каждый бит

показывает, находится ли определенный предмет в рюкзаке или нет(0 – нет, 1 – есть). Шифротекстом будет являться сумма всех находящихся в рюкзаке предметов. Данная сумма затем передается по каналу связи. Криптосистема Меркла – Хеллмана является ассиметричной. Это означает, что для шифрования и расшифрования используются разные ключи. Эти ключи математически тесно связаны. Рюкзачным вектором будем называть вектор $A = (a_1, \ldots, a_n)$ – упорядоченный набор из n предметов, где $a_i, i = 1 \ldots n$ – вес i – того предмета.

При расшифровке можно столкнуться с одной из двух проблем: «лёгкой» или «трудной». Легкая проблема предполагает, что рюкзачный вектор – сверхрастущий(т.е. каждый новый элемент вектора больше суммы всех предыдущих). В таком случае расшифровка тривиальна и выполняется за конечное время. В случае «трудной» проблемы, рюкзачный вектор сверхрастущим не является. Для данной проблемы решение – перебор всех значений, что намного затратнее по времени в сравнении с лёгкой проблемой. На время решения данной задачи сильно влияет количество элементов в рюкзачном векторе. Например, для последовательности из шести элементов нетрудо найти решение, даже если проблема «трудная». Реальные рюкзаки должны содержать не менее 250 элементов. Длина каждого члена сверхвозрастающей последовательности должна быть где-то между 200 и 400 битами, а длина модуля – от 100 до 200 бит. При таких условиях проверка всех возможных вариантов займет свыше 10^{46} лет[8].

После того, как эта криптосистема увидела свет, другие специалисты в области криптографии стали разрабатывать и предлагать свои криптосистемы, основанные на задаче о рюкзаке. Были созданы рюкзаки Грэма – Шамира, Накаше – Штерна, Шор – Ривеста, и т.д. Большинство из них на сегодняшний призаны небезопасными.

Рассмотрим класс криптосистем с рюкзаком, обладающим заданными свойствами; также введем обозначения, которые пригодятся в будущем.

2 Обобщенный и нестандартный рюкзаки

Криптостойкость рюкзачных криптосистем зависит от способа кодирования сообщений, в частности самих букв и от процедуры последующего шифрования текста[6]. Рассмотрим класс криптосистем с открытым ключом и рюкзаком, обладающим заранее заданными свойствами.

Пусть задан набор

$$\mathbf{A} = (\mathbf{a_1}, \mathbf{a_2}, \dots \mathbf{a_n}) \quad \textbf{(3)}$$

– рюкзачный вектор размерности n, $n \geq 3$ из n натуральных чисел a_i, $i = 1 \dots n$ и некоторое натуральное число $v \geq 0$. Необходимо определить, существуют ли такие значения $\alpha_i \in \{0,1\}$, которые удовлетворяют равенству

$$\sum_{i=1}^{n} \boldsymbol{\alpha_i}\, \mathbf{a_i} = \mathbf{v} \quad \textbf{(4)}$$

Пару (A, v) с определенными A и v будем называть входом. Для любого входа (A, v) найти такие значения α_i либо невозможно, либо существует несколько подходящих последовательностей α_i длины n, либо существует единственный подходящий вариант. Такой вариант однозначно определяется для каждого входа (A, v), если рюкзак является сверхрастущим. Для сверхрастущего рюкзака выполняется свойство:

$$\mathbf{a_j} > \sum_{k=1}^{j-1} \mathbf{a_k}, \mathbf{j} = \mathbf{2} \dots \mathbf{n} \quad \textbf{(5)}$$

Будем называть рюкзачный вектор с повторениями, если его компоненты повторяются и без повторений, если они не повторяются.

Далее введем обозначения:

$$\boldsymbol{ZK_t = \{k_1, k_2, \dots k_t\}, k_1 + k_2 + \dots + k_t = n,} \quad \textbf{(6)}$$

– множество коэффициентов повторений компонентов рюкзачного вектора и

$$\mathbf{ZC_p = \{m_1, m_2, \dots, m_n\}} \quad \textbf{(7)}$$

– множество коэффициентов повторений компонентов входа (A, v).

Здесь элемент $k_i, k_i \geq 1, i = 1 \dots t$ – количество повторений компонентов ранга числа a_i в рюкзаке(причем t – количество разных компонентов рюкзачного вектора), а m_i – максимальное значение коэффициента α_i при a_i в рюкзаке A для определения входа (A, v), причем $0 \leq m_i \leq p - 1, i = 1 \dots n, p \geq 2, p \in N$. Множества ZK_t и ZC_p назовем спектрами коэффициентов рюкзака (1) и его входа соответственно.

Для стандартной рюкзачной криптосистемы решением для входа (A, v) является подмножество элементов из A, такое, что выполняется равенство (2), причем $\alpha_i \in \{0,1\}$. Но если $\alpha_i \in \{0,1, \dots, m_i\}$, i = 1…n, и хотя бы один из $\alpha_i \geq 2$, то решение – подмножество элементов из A с коэффициентами повторений компонентов из ZC_p. Это означает, что существует вектор $w_v = (\alpha_1, \alpha_2, \dots, \alpha_n)$ представляющий собой спектр повторений для v, где $\alpha_i \in \{0,1, \dots, m_i\}$, при котором выполняется

$$\mathbf{v = A \cdot w_v^T} \quad \textbf{(8)}$$

Причем решение будет без повторений или с повторениями, если соответственно p = 2 или p > 2. Мощностью входа будем называть

количество всех допустимых числовых значений v для рюкзачного вектора A. Обозначим её через $\mu(V_A)$, где $V_A = \{v_0, v_1, \dots, v_t\}$ – множество значений v_I, для которых входы (A, v_i) имеют решения. Но т.к. для одного v может быть несколько решений, то через $\mu(A)$ обозначим мощность рюкзака. Рюкзак будем называть инъективным, если для каждого входа существует одно единственное уникальное решение.

Пусть, $a_i \in ZK_t$ может повторяться и входить в сумму для определения входа (A, v) с коэффициентом $\alpha_i \leq m_i \in ZC_p, i = 1 \dots n$. Тогда такой рюкзачный вектор будем называть нестандартным и обозначать $\widetilde{A}$. Если для такого рюкзачного вектора все коэффициенты из ZK_t = 1, как и коэффициенты из ZC_p, то речь идет об стандартном рюкзаке. Если же все коэффициенты из ZC_p = p – 1, то мы имеем дело с обобщенным рюкзаком с заданным максимальным числом повторений всех его компонент = p – 1.

С точки зрения криптографии, криптографические системы, построенные на задачах обобщенного и нестандартного рюкзаков на порядок криптоустойчивее в сравнении со стандартным рюкзаком. Действительно, если обозначить через A_2, A_p, и $\widetilde{A}$ стандартный рюкзак, обобщенный рюкзак и нестандартный рюкзак соответственно, то количество всех вариантов выбора ключей для стандартного рюкзака – $N_{A_2}(K) = 2^n$, для обобщенного рюкзака - $N_{A_p}(K) = p^n$, а для нестандартного рюкзака $N_{\widetilde{A}}(K) = (m_1 + 1)(m_2 + 1) \dots (m_n + 1)$, где $m_i \in ZC_p$.

Исходя из определений, для обобщенного рюкзака ключом будет являться вектор $w_v = (\alpha_1, \alpha_2, \dots, \alpha_n)$, т.е. по факту, это целочисленная последовательность, в которой цифры не превышают p – 1. Рассмотрим вопрос выбора таких последовательностей для шифрования сообщений с помощью кода Варшамова.

3 Код Варшамова

Пусть для данного канала информации задан алфавит $B = \{0,1,\dots,p-1\}$, а также a – произвольное неотрицательное целое число, n – длина кодового слова $X = x_1x_2\dots x_n,\ x_i \in B, i = 1\dots n.$ Тогда мы получим[9] множество слов, для которых выполняется следующее сравнение:

$$\boldsymbol{W = \sum_{i=1}^{n} ix_i \equiv a(mod\ n+1)} \qquad \textbf{(9)}$$

Данное множество слов и образует код Варшамова, т.е.

$$W_n = \{x_1x_2\dots x_n | W = \sum_{i=1}^{n} ix_i \equiv a(mod\ n+1), x_i \in B, i = 1\dots n\} \qquad \textbf{(10)}$$

Так, в частности, при p = 3, n = 4, a = 0 из указанного сравнения получим код W_4, состоящий из 17 кодовых слов:

0 0 0 0	1 1 1 1	0 2 1 2	0 1 0 2	0 1 1 0	2 0 1 0
1 0 0 1	2 1 2 0	0 2 2 0	0 0 2 1	1 0 2 2	2 2 0 1
1 2 0 0	2 0 0 2	1 2 2 1	2 1 1 2	2 2 2 2	

Таблица 1 - Код Варшамова

Задавая другие параметры p, n, и a, можно получить другие результаты. К примеру, при p = 2, n = 8, a = 0 имеем 30 кодовых слов, удовлетворяющих сравнению (10), но если взять p = 6, n = 8, a = 0, то можно получить 186624 кодовых слов, удовлетворяющих сравнению (10). Мощность кода Варшамова для двоичного случая вычисляется по формуле

$$|\boldsymbol{W_n}| = \frac{1}{2(n+1)} \sum_{d|n+1} \varphi(d)\, 2^{(n+1)/2} \qquad \textbf{(11)}$$

где $\boldsymbol{\varphi(d)}$ – количество чисел i ($1 \leq i \leq d$), взаимно простых с d.

4 Применение кодов Варшамова к обобщенному рюкзаку

Как говорилось ранее в части 2, для обобщённого рюкзака ключом будет являться вектор $w_v = (\alpha_1, \alpha_2, \dots, \alpha_n)$ – целочисленная последовательность, в которой цифры не превышают p – 1. Рассмотрим связь кода Варшамова и коэффициентов $\alpha_1, \alpha_2, \dots, \alpha_n$.

Введем следующее сравнение[9]:

$$\boldsymbol{W_{\alpha,\beta}} = \sum_{i=1}^{n} \beta_i \alpha_i \equiv a(mod\ m), \tag{12}$$

где $\boldsymbol{\beta_i}$ – члены заданной числовой последовательности, $\boldsymbol{\alpha_i}$ могут принимать значения из различных числовых множеств. В частности, совокупность всех бинарных решений сравнения при $\boldsymbol{\beta_i} = i, m = 2n, a \geq 0$ задает код Варшамова, исправляющий канальные одиночные вставки. Но если взять в качестве β рюкзачный вектор, а в качестве α – спектр некоторого элемента открытого текста, то получим систему защиты информации.

Пусть $m = m_1, m_2, \dots, m_n$ – множество всех элементарных сообщений, каждое из которых может быть либо одним символом, либо их конкатенацией, $W = W_1, W_2, \dots, W_k$ – кодовые слова из кода Варшамова.

Для случая стандартного рюкзака схема была следующей: рюкзак либо содержал определенный предмет, либо нет. Исходя из этого, вектор w_v состоял из нулей и единиц, в соответствии с определением стандартного рюкзака. Таким образом, эта двоичная последовательность представляла собой запись какого – то числа в двоичной системе счисления. В свою очередь, это число соотвествовало элементарному сообщению из m. Это может быть как символ, так и их конкатенация.

Таким образом, элементарное сообщение представлялось в виде нулей и единиц в качестве вектора w_v.

Случай обобщенного рюкзака предполагает несколько иной подход: в данной ситуации координатами вектора w_v могут являться любые числа от 0 и до p – 1 включительно. Практически, это означает что возможным решением задачи обобщенного рюкзака может быть ситуация, когда какой то определенный предмет входит в рюкзак несколько раз, но не более чем p раз. Тогда данная последовательность – запись некоторого числа в p – ой системе счисления, которое, в свою очередь, соответствует элементарному сообщению из m.

Отдельные элементарные сообщения и код Варшамова можно связать следующим образом: Каждому элементарному сообщению из m будем ставить в соответствие некоторое кодовое слово W_i из кода Варшамова с определенными параметрами p, n и a. В зависимости от этих параметров, мы можем получить разные кодовые слова. Из общего множества данных кодовых слов мы можем выбрать те, которые будут соответствовать элементарным сообщениям из m, причем каждому такому сообщению будут соответствовать разные кодовые слова из кода Варшамова. Смысл заключается в следующем: использование кода Варшамова позволяет значительно повысить криптоустойчивость системы защиты информации.

Используя данные коды, можно самому определить, какое сообщение будет соответствовать очередному кодовому слову из всего множества. Пусть $|m|$ – мощность множества всех сообщений. Подразумевается, что $|m| \leq |W|$, где $|W|$ – мощность кода Варшамова с определенными параметрами p, n, и a. В противном случае, установка однозначного соответствия между отдельными сообщениями из m и кодовыми словами невозможна.

Если смотреть на данную проблему выбора кодовых слов с точки зрения криптографии, то данный подход в сравнении со случаем стандартного рюкзаком позволяет значительно усложнить расшифровку криптограммы, при условии, что ключ расшифровки неизвестен. Как уже упоминалось в п.2, количество всех вариантов выбора ключей для стандартного рюкзака – $N_{A_2}(K) = 2^n$, а для обобщенного рюкзака - $N_{A_p}(K) = p^n$. Простой перебор при больших p и n значительно затруднит взлом.

Дополнительной сложностью на пути взломщика станет таблица соответствий элементарных сообщений и кодовых слов из кода Варшамова. Для однозначной расшифровки необходимо знать, какое из кодовых слов соответствует определенному сообщению из m. Если, к примеру, взять случай двоичного рюкзака: предположим, что взломщик нашел решение задачи, т.е. нашел подходящую последовательность $w_v = (\alpha_1, \alpha_2, \dots, \alpha_n)$, которая удовлетворяет соотношению (4). Далее он преобразует эту двоичную последовательность в сообщение из m, получив часть от изначального сообщения. Но при наличии таблицы соответствия задача усложняется тем, что банальный перевод из двоичной системы счисления в текст при помощи, например, таблицы ASCII – не подойдет. Каждому такому сообщению соответсвует определенное, уникальное, выбранное заранее кодовое слово. В итоге перебор различных последовательностей типа $w_v = (\alpha_1, \alpha_2, \dots, \alpha_n)$ без таблицы соответствий если и даст результат, то он в общем случае будет неверным.

Рассмотрим математическую модель криптосистемы на основе обобщенного рюкзака с применением кода Варшамова.

5 Математическая модель криптосистемы на основе обобщенного рюкзака с применением кода Варшамова

Криптосистема – общий комплекс мер по защите информации от несанкционированного доступа, порчи данных и замены. Определим математическую модель для рассматриваемой задачи:

$$\Delta = \langle \mathrm{M}^*, \mathrm{S}^*, \mathrm{E(m)}, \mathrm{D(s)}, W(p,n,a), T(\mathrm{M}^*, W) | \ \mathrm{V(E(m), D(s)}, T(\mathrm{M}^*, W)) \rangle \quad (13)$$

Разберем элементы модели подробнее. В соответствии с [10], M^* – множество всех сообщений $\mathrm{m} = \mathrm{m}_1, \mathrm{m}_2, \ldots, \mathrm{m}_\mathrm{n}$ (открытых текстов) над буквенным/числовым алфавитом M (непустое упорядоченное множество символов). Здесь m_i, $\mathrm{i} = 1 \ldots \mathrm{n}$ – множество элементарных сообщений. Ими могут быть как отдельные буквы, так и конкатенация букв из алфавита M. S^* - множество всех криптограмм $\mathrm{s} = \mathrm{s}_1, \mathrm{s}_2, \ldots, \mathrm{s}_\mathrm{n}$ системы. $\mathrm{E(m)}$ – алгоритм прямого преобразования сообщения m в шифротекст s; $\mathrm{D(s)}$ – алгоритм обратного преобразования шифротекста s в сообщение m.

Дополнительно определим $\boldsymbol{W(p,n,a)}$ – код Варшамова с параметрами p, n, a. $\boldsymbol{T(\mathrm{M}^*, W)}$ – таблица соответствия кодовых слов из кода Варшамова $\boldsymbol{W}$ и сообщений из M^*.

Алгоритмы $\mathrm{E(m)}$ и $\mathrm{D(s)}$, а также $\boldsymbol{T(\mathrm{M}^*, W)}$ связаны между собой при помощи $\mathrm{V(E(m), D(s)}, \boldsymbol{T(\mathrm{M}^*, W)})$. Данная связь подразумевает, что любое сообщение $\mathrm{m} \in \mathrm{M}^*$ однозначно преобразовывается в шифротекст $\mathrm{s} \in \mathrm{S}^*$ при помощи $\mathrm{E(m)}$ и таблицы соответствия $\boldsymbol{T(\mathrm{M}^*, W)}$, а любой шифротекст $\mathrm{s} \in \mathrm{S}^*$ однозначно преобразовывается в оригинальное сообщение $\mathrm{m} \in \mathrm{M}^*$ при помощи $\mathrm{D(s)}$ и той же таблицы соответствия $\boldsymbol{T(\mathrm{M}^*, W)}$.

6 Криптосистема на основе обобщенного рюкзака с применением кода Варшамова

Основа данной криптосистемы – задача об обобщенном рюкзаке. Его мы рассматривали в части 2. Повторим основную информацию:

Пусть задан набор

$$\mathbf{A} = (\mathbf{a_1}, \mathbf{a_2}, \dots \mathbf{a_n}) \quad \textbf{(14)}$$

– рюкзачный вектор размерности n, $n \geq 3$ из n натуральных чисел a_i, $i = 1 \dots n$ и некоторое натуральное число $v \geq 0$. Необходимо определить, существуют ли такие значения $\alpha_i \in \{0,1, \dots p-1\}$, которые удовлетворяют равенству

$$\sum_{i=1}^{n} \boldsymbol{\alpha_i}\, \mathbf{a_i} = \mathbf{v} \quad \textbf{(15)}$$

Как и в случае с алгоритмом Меркла – Хеллмана, данный алгоритм - ассимметричный: Для шифрования и расшифрования используется пара ключей – открытый и закрытый. Оба ключа тесно связаны между собой математически, что позволяет получить один ключ из другого и наоборот.

Будем называть $A_p = (a_1, \dots, a_n)$ упорядоченным набором из n предметов, причем эти предметы могут входить в рюкзак не более p – 1 раз. Выберем два числа, m и q так, что m подчиняется сравнению:

$$m > \sum_{i=1}^{n} a_i \,, m \in N \quad \textbf{(16)}$$

В свою очередь, для числа q должно выполняться условие:

$$q < m, \mathrm{q} \in \mathrm{N} \quad \textbf{(17)}$$

Наибольший общий делитель этих двух чисел должен быть равен 1. Это необходимо для того, чтобы для числа q существовало мультипликативно обратное по модулю m число u, такое, что выполняется равенство:

$$\mathrm{q} * \mathrm{u}(\mathrm{mod}\ \mathrm{m}) = 1 \quad \textbf{(18)}$$

Также определим вектор $B = (b_1, b_2, \dots, b_n)$, полученный из A_p сильным модульным умножением при помощи следующего соотношения:

$$\mathrm{B} = \boldsymbol{A_p} * q(mod\ m) \quad \textbf{(19)}$$

Если говорить о сверхрастущем векторе A_p, то в сравнении с алгоритмом Меркла – Хеллмана условие для данной криптосистемы несколько иное.

– Для криптосистемы на двоичном рюкзаке:

Рюкзачный вектор A называется сверхрастущим, если:

$$\sum_{\mathbf{i=1}}^{\mathbf{j-1}} \mathbf{a_i} < \mathbf{a_j}, \mathbf{j} = \mathbf{2}, \dots, \mathbf{n} \quad \textbf{(20)}$$

– Для криптосистемы на обобщённом рюкзаке:

Рюкзачный вектор A_p называется сверхрастущим, если:

$$\sum_{i=1}^{j-1} a_i \alpha_i < a_j, j = 2, \ldots, n, \alpha_i \geq p - 1 \; \forall i = 1 \ldots j - 1 \qquad (21)$$

Условие (19) – условие однозначного шифрования и дешифрования для криптосистемы на обобщенном рюкзаке[11].

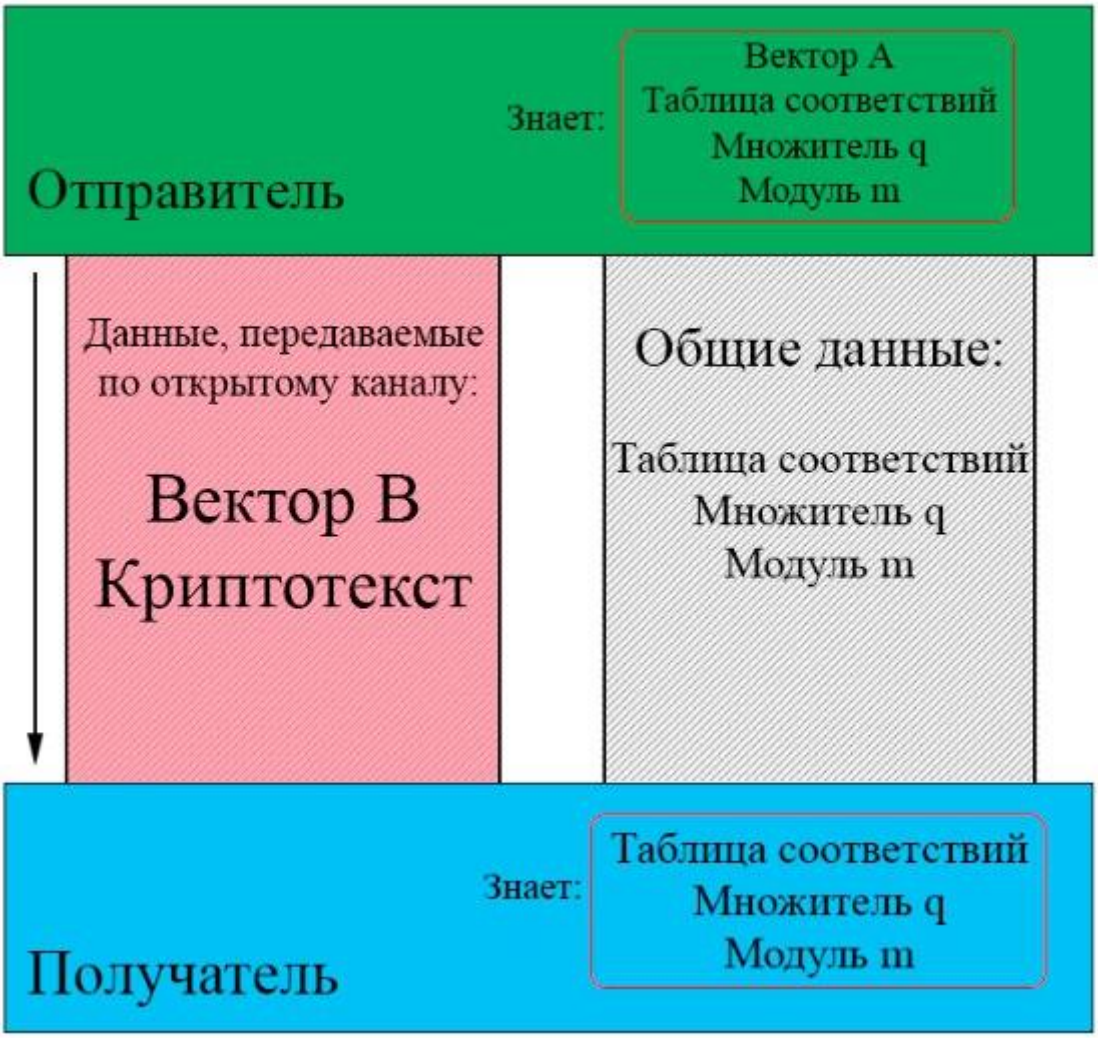

Рисунок 1 – Схема «Получатель - Отправитель»

Разберем алгоритм шифрования на основе этой задачи.

Для того, чтобы зашифровать некоторый текст, необходимо:

– Сгенерировать код Варшамова таким образом, чтобы каждое кодовое слово было равным по длине с длиной рюкзачного вектора A_p(т.е.

n должен совпадать с количеством элементов в рюкзачном векторе.), а параметр p кода Варшамова совпадал с параметром p рюкзачного вектора A_p.

– Для указанного кода Варшамова и необходимо создать таблицу соответствия, которая будет связывать отдельные элементарные сообщения и кодовые слова из кода Варшамова.

– Для каждого элементарного сообщения вычислить:

$$F_{A_p}(X) = A_p * W_x \quad \textbf{(22)}$$

где A_p – рюкзачный вектор, W_x – кодовое слово элементарного сообщения X. Результат вычисления – сумма.

Полученный набор сумм и будет являться криптотекстом.

Для того, чтобы расшифровать криптотекст, необходимо:

– Восстановить вектор A из вектора B при помощи мультипликативно обратного к множителю q по модулю m следующим образом:

$$A = B * u(\mathrm{mod}\ m) \quad \textbf{(23)}$$

где u – мультипликативно обратное по модулю m.

– Для каждой суммы v_i из криптограммы найти такой $\boldsymbol{W_x}$, что выполняется равенство:

$$A_p * W_x = v_i \quad \textbf{(24)}$$

– Сопоставить каждому полученному $\boldsymbol{W_x}$ элементарное сообщение, кодируемое им, получая в итоге исходное сообщение.

Разберем пример работы алгоритма[11]:

Пусть T = CODE RRW BAC AB AB – исходный текст. Для p = 3, n = 6, a = 0 получим код W_6, состоящий из 105 слов.

Вот первые 30 кодовых слов:

000000	000112	000120	000201	001011	001100
001212	001220	002022	002111	002200	010002
010010	010122	010211	011021	011102	011110
011222	012001	012121	012202	012210	020012
020020	020101	020221	021000	021112	021120

Таблица 2 – 30 кодовых слов из кода W_6

Далее выбираем случайным образом 26 слов из W_6 для первоначального способа кодирования. Каждое выбранное слово будет соответствовать одной букве из английского алфавита.

Таблица 3 – таблица соответствия кодовых слов и букв алфавита

A-000000	L-010211	B-000120	C-000201	D-001011	E-001100
F-001212	U-020020	G-002022	H-002111	I-002200	J-010002
K-010010	Z-021201	Y-021120	M-011021	N-011102	O-011110
P-011222	X-021112	Q-012121	R-012202	S-012210	T-020012
W-021000	V-020221	_-100001			

Выберем в качестве рюкзачного вектора вектор $A_3 = (1,2,6,11,21,43)$. Определив функцию шифрования как (22), получим:

Таблица 4 – таблица кодировок символов

$F_{A_3}(\ \_\ ) = (1,2,6,11,21,43) * (100001) = 44$
$F_{A_3}(\ A\ ) = (1,2,6,11,21,43) * (000000) = 0$
$F_{A_3}(\ B\ ) = (1,2,6,11,21,43) * (000120) = 53$
$F_{A_3}(\ C\ ) = (1,2,6,11,21,43) * (000201) = 65$
$F_{A_3}(\ D\ ) = (1,2,6,11,21,43) * (001011) = 70$
$F_{A_3}(\ E\ ) = (1,2,6,11,21,43) * (001100) = 17$

Продолжение таблицы 4

$F_{A_3}(\ O\ ) = (1,2,6,11,21,43) * (011110) = 40$
$F_{A_3}(\ R\ ) = (1,2,6,11,21,43) * (012202) = 122$
$F_{A_3}(\ W\ ) = (1,2,6,11,21,43) * (021000) = 10$

В этом случае исходный текст шифруется как:

$$\text{E} = \mathbf{65, 40, 70, 17, 44, 122, 122, 10, 44, 53, 0, 65, 44, 0, 53, 44, 0, 53} \quad \textbf{(25)}$$

В примере параметры p, n, и a подобраны так, что $|W_6|$ = 105 > 26. Очевидно, что код, мощность которого меньше чем количество букв в алфавите, нам не подойдет. К тому же, для однозначного шифрования/расшифрования данный рюкзачный вектор не подойдет, в соответствии с (21). Легальный получатель может иметь полученные подстановки и наиболее простой вариант рюкзачного вектора, которые являются лазейкой, известной только легальным пользователям системы.

Если для вектора $A_3 = (2,5,15,45,135,405)$ взять секретную пару m = 2003 и q = 738 (u = 19), то тогда открытый ключ - $B_3 = (1476,1687,1055,1162,1483,443)$. Этот вектор уже не является сверхрастущим, поэтому криптоаналитик столкнется с большими сложностями при криптоанализе данного вектора. Но это не помеха для легального получателя, т.к. он применит к полученному вектору B_3 секретный ключ u = 19 и получит основной ключ $A_3 = (2,5,15,45,135,405)$, которым и расшифрует криптограмму.

7 Реализация криптосистемы на основе обобщенного рюкзака с применением кода Варшамова

Поставленная задача: разработать систему защиты информации на основе обобщенного рюкзака с применением кодов Варшамова. Данный алгоритм реализован на языке Python с использованием PyQt – набора привязок графического фреймворка Qt для языка программирования Python. Также использовались библиотеки:

– NumPy – библиотека, необходимая для произведения научных вычислений, работы с n – мерными массивами и преобразованиями в линейной алгебре;

– Random – встроенная в Python библиотека, позволяющая работать со случайными значениями.

Целью работы была реализация алгоритма шифрования/расшифрования в виде программы, способной:

– Работать с файловой системой;

– Вести журнал событий;

– Предоставлять инструмент генерации закрытого и открытого ключей, а также кода Варшамова с указанными параметрами;

– Предоставлять возможность ручного ввода открытых и закрытых ключей;

– Производить шифровку и расшифровку сообщений в рамках алгоритма.

Рассмотрим детали интерфейса в подробностях.

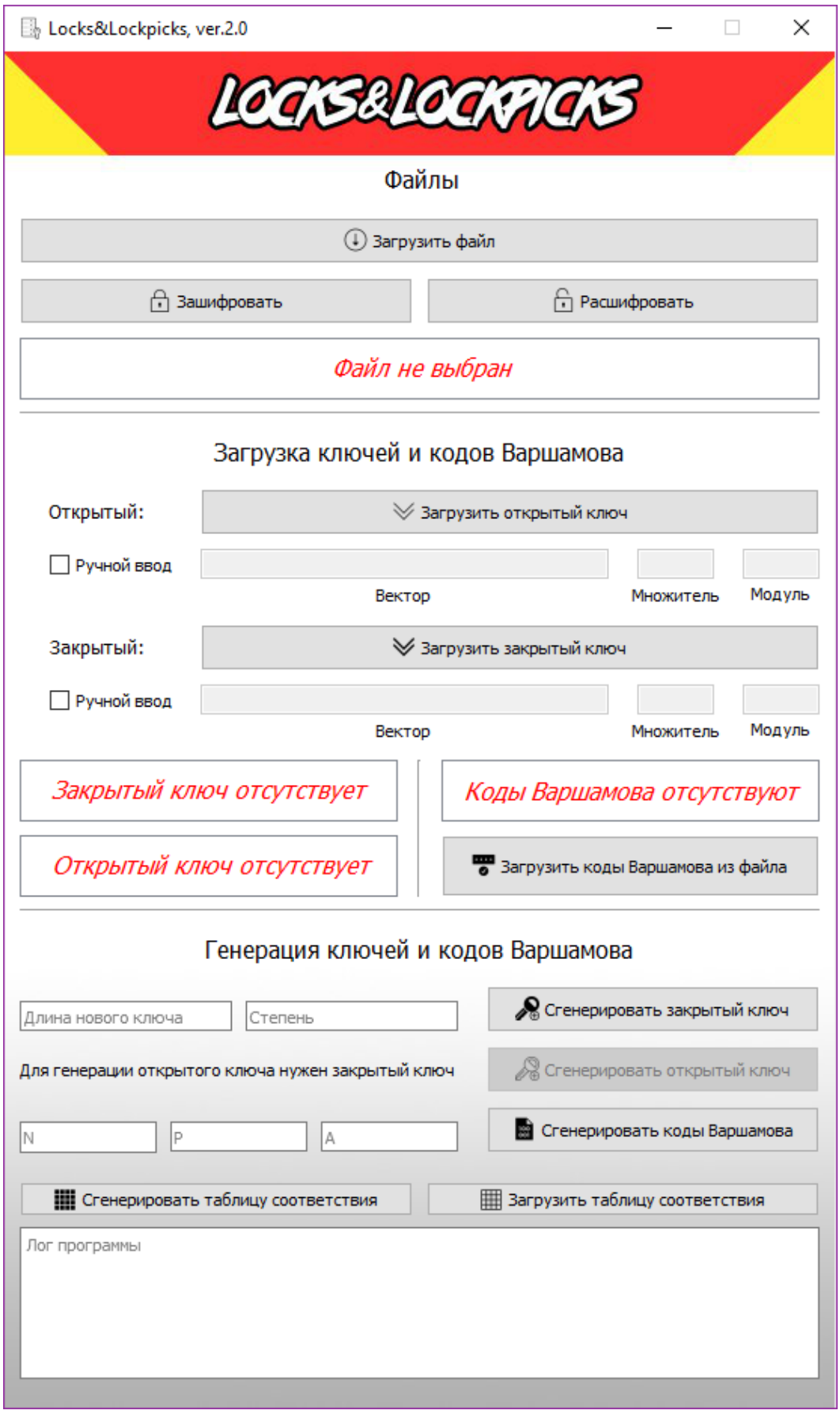

Рисунок 2 – основной интерфейс программы

Верхняя область окна отвечает за работу с файлами, содержащими исходный текст или криптограмму. Кнопка «Загрузить файл» необходима для выбора файла, который необходимо обработать. Подходящими файлами являются файлы, содержащие в себе цифры от 0 до 9, заглавные буквы английского алфавита, а также символы «.», «,», «!», «?», «\n». Данное множество символов образует алфавит, который использует программа для шифрования/расшифрования. Кнопка «Зашифровать» запускает процедуру зашифровки выбранного файла. Однако, для самого процесса зашифровки необходимы дополнительные данные, помимо самого файла с текстом. Этими данными являются:

– Закрытый ключ(рюкзачный вектор A_p)

– Таблица соответствий кодовых слов из кода Варшамова и отдельных символов

Без этих данных процедура не будет запущена. Узнать о том, присутствуют ли эти данные в системе или нет, можно в логе программы, находящемся в нижней части интерфейса. Для закрытого ключа присутствует отдельный индикатор, свидетельствующий о состоянии ключа(загружен/отсутствует). Если не все условия соблюдены, то в лог будет выведено уведомление.

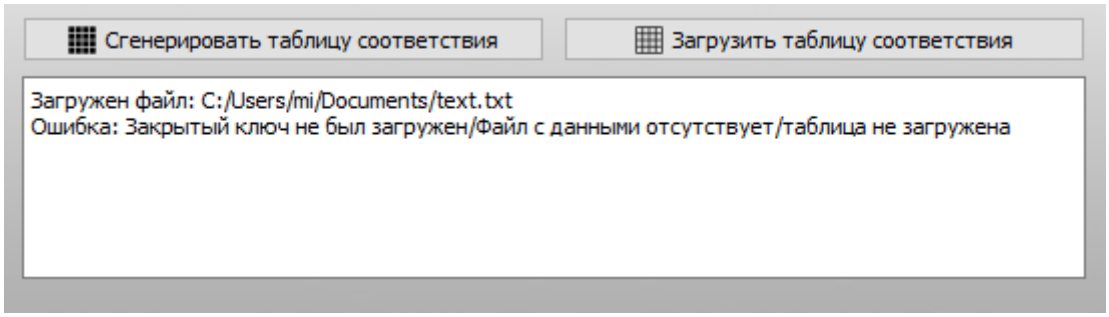

Рисунок 3 – лог программы при неудачной операции шифровки

Ситуация аналогична для кнопки «Расшифровать». Для расшифровки требуется загрузить файл с криптограммой, открытый ключ, а также таблицу соответствий, которая использовалась при зашифровании

файла. При использовании другой таблицы результат расшифровки может весьма отличаться от исходного текста.

Центральная область окна отвечает за загрузку ключей и загрузку кода Варшамова из файла. Также там присутствуют индикаторы, отображающие состояние открытого, закрытого ключей, а также кодов Варшамова. Кнопки «Загрузить открытый ключ» и «Загрузить закрытый ключ» позволяют выбрать файл, содержащий открытый и закрытый ключи соответственно. Однако в программе предусмотрена функция ручного ввода ключей: при желании пользователь может активировать «Ручной ввод». В таком случае, пользователь может ввести вектор, множитель и модуль самостоятельно в специально отведенные для этого поля. Компоненты вектора вводятся через запятую. При нажатии на кнопку загрузки ключа вместо открытия окна выбора файла программа считает введенные данные и загрузит их.

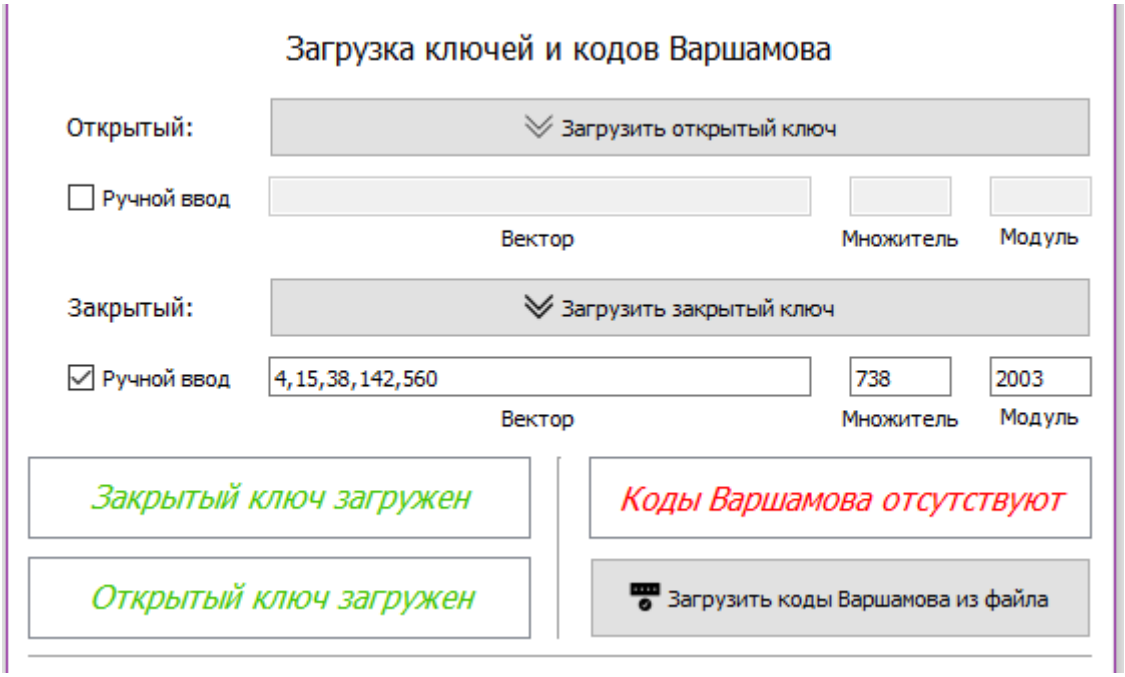

Рисунок 4 – окно после загрузки ключей

В нижней части окна находится генератор открытых и закрытых ключей. Для закрытого ключа необходимо указать его длину, а также параметр p. После нажатия на кнопку «Сгенерировать закрытый ключ» программа создаст новый ключ, загрузит его в программу и предложит

сохранить в файл. Для генерации открытого ключа необходим загруженный заранее открытый ключ. После генерации, также как и для загрытого ключа, он будет загружен, а затем будет предложено сохранить полученный ключ. Для генерации кодов Варшамова требуется три параметра: p, n и a в соответствии с (10). Также, как и при создании ключей, при генерации кодовых слов полученное множество будет автоматически загружено, а затем откроется окно для сохранения полученных кодов.

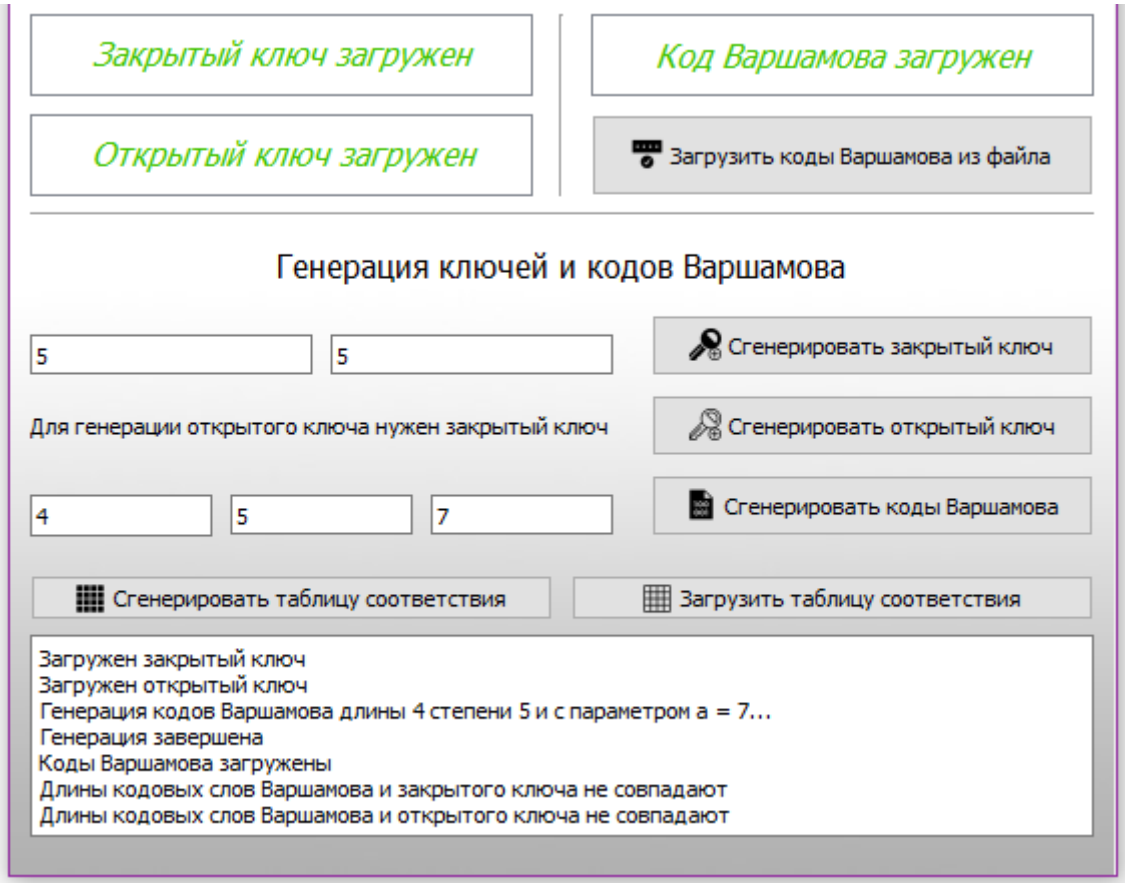

Рисунок 5 – окно программы после генерации ключей и кодов Варшамова

Важно отметить, что для корректной работы алгоритма необходимо, чтобы длина кодовых слов совпадала с длиной закрытого ключа(т.е. рюкзачного вектора), в соответствии с (22). Также, необходимо, чтобы было выполнено условие (21) для однозначного шифрования/дешифрования. Закрытый ключ в комбинации с кодами

Варшамова образует таблицу соответствия, необходимую для представления символов алфавита в виде кодовых слов с заданной длиной n и параметрами p и a. Данная таблица необходима при шифровании сообщения и расшифровке криптограммы.

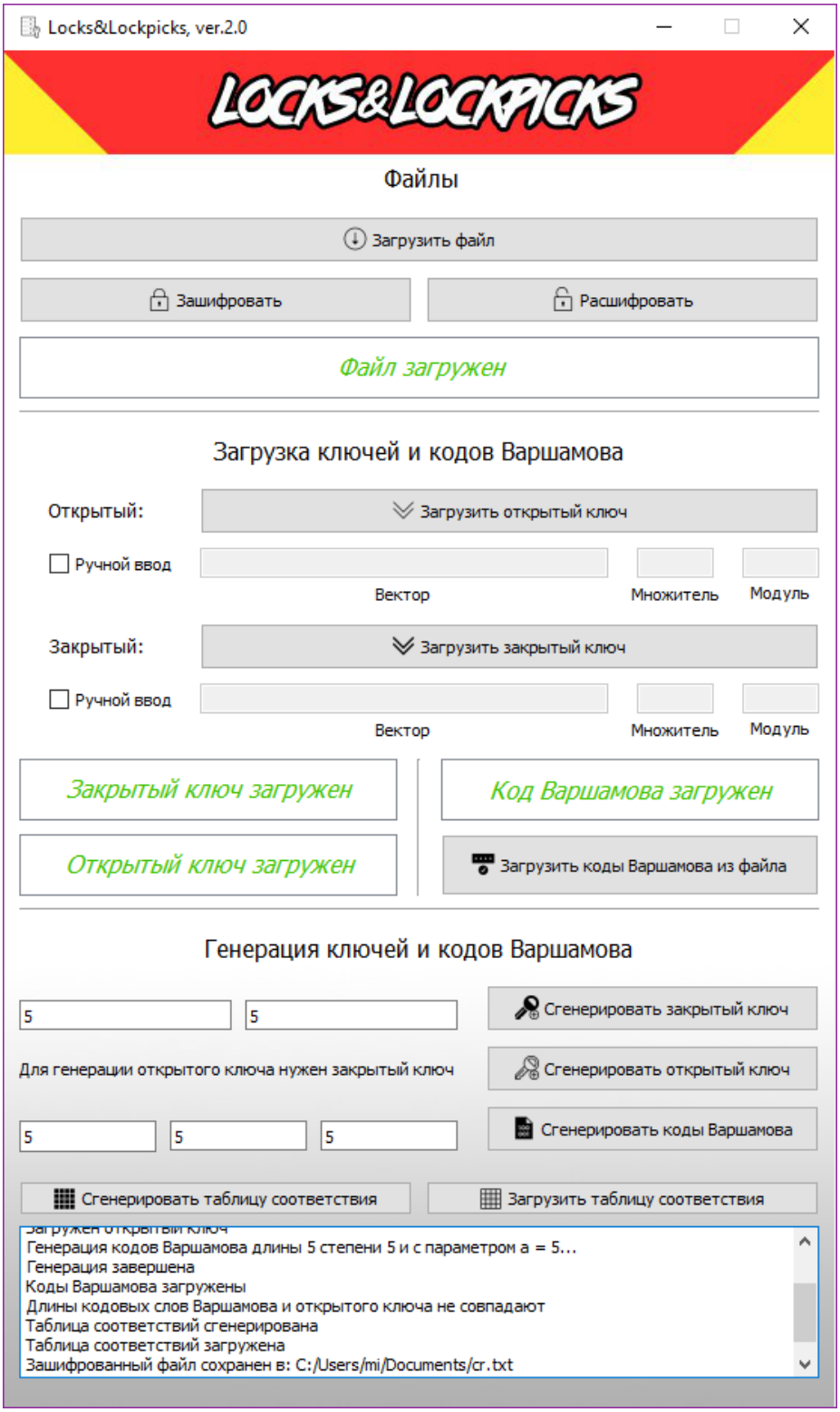

Рисунок 6 - окно программы после успешого зашифрования

ЗАКЛЮЧЕНИЕ

Нами была разработана математическая модель криптосистемы на основе обобщенного рюкзака с применением кодов Варшамова, способная шифровать и расшифровывать файлы данных, а также генерировать необходимые для работы ключи, коды Варшамова и таблицу соответствия. В сравнении с алгоритмами, основанными на симметричном шифровании, данный алгоритм работает медленнее. Но несмотря на это, алгоритм предоставляет большую криптостойкость в сравнении с алгоритмом Меркла – Хеллмана. Данная реализация криптосистемы является модификацией СЗИ на основе двоичного рюкзака. Криптоустойчивость такой системы на порядок выше криптосистемы на основе двоичного рюкзака.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 Левитин А. В. Алгоритмы. Введение в разработку и анализ — М.: Вильямс, 2006. — 576 с.

2 Silvano Martelo, Paolo Toth. Knapsack problems. — Great Britain: Wiley, 1990. — 306 с. — ISBN 0-471-92420-2.

3 Kellerer H., Pferschy U., Pisinger D. Knapsack Problems — Springer Science+Business Media, 2004. — 548 с. — ISBN 978-3-642-07311-3

4 G. Gallo, P. L. Hammer, B. Simeone Quadratic knapsack problems // Mathematical Programming Studies. — 2009. — 24 февраль (vol. 12). — P. 132-149. — ISSN 0303-3929

5 Осипян, В.О. Построение систем защиты информации на основе проблемы универсального рюкзака / В.О. Осипян, Е.А. Семенчин // Известия ТРТУ. — С. 182-188.

6 Осипян, В.О. О системе защиты информации на основе проблемы рюкзака // Известия Томского политехнического университета. – 2006. – Т. 309. – № 2. –С. 209-212.

7 Ralph Merkle and Martin Hellman, Hiding Information and Signatures in Trapdoor Knapsacks // Information Theory. — 1978. —Т. 24. — №5. — С. 525–530.

8 Шнайер, Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си [Текст] / Б. Шнайер — М.: Триумф, 2002. — 816 с.

9 Осипян, В.О. Об одной кодовой криптосистеме на основе кода Варшамова / В.О. Осипян, В.В. Подколзин, А.А. Арджанов//Решетневские чтения. —2009. — С. 568-569.

10 Осипян, В.О. Математическая модель системы защиты информации на основе диофантова множества / В.О. Осипян, А.В.

Мирзаян, Ю.А. Карпенко, А.С. Жук, А.Х. Арутюнян // Чебышевский сборник. — 2014. —Т. 15. —№ 1. — С. 146-154.

11 Осипян, В.О. Система защиты информации на основе кода Варшамова // Информационное противодействие угрозам терроризма. — 2003. —№ 1. — С. 121-123.