Учреждение образования

«Белорусский государственный технологический университет»

**Кафедра информатики и веб-дизайна**

**Лабораторная работа №10**

Исследование асимметричных шифров RSA и Эль-Гамаля

Выполнил:

Студент 3 курса 10 группы ФИТ

Баранчук Владислав

2021 г.

1. **Теоретическая часть**

Алгоритм RSA появился (1977 г.) после алгоритма рюкзака Меркла. Он стал первым полноценный алгоритмом с открытым ключом, который впоследствии стал одним из основных для шифрования и для электронных цифровых подписей.

Из всех предложенных алгоритмов с открытыми ключами RSA проще всего понять и реализовать. Названный в честь трех его создателей: Рона Ривеста (RonRivest), Ади Шамира (Adi Shamir) и Леонарда Эдлемана (Leonard Adleman).

Как было отмечено, безопасность RSA основана на трудности разложения на множители больших чисел. Открытый и закрытый ключи являются функциями двух больших простых чисел. Предполагается, что восстановление открытого текста по шифртексту и открытому ключу эквивалентно разложению на множители двух больших чисел.

Для генерации двух ключей: тайного и открытого (а по сути – двух взаимосвязанных частей одного ключа, т. е. ключа, принадлежащего одному физическому лицу (или группе лиц), либо одному юридическому лицу) используются два больших случайных простых числа, p и q. Для максимальной большей криптостойкости нужно выбирать p и q равной длины.

Рассчитывается произведение: n = pq. Этой есть один из трех компонент

ключа, состоящего из чисел n, e, d. Затем случайным образом выбирается второй компонент ключа (открытый ключ или ключ зашифрования, e, такой что e и (p-1)(q-1) являются взаимно простыми числами; вспомним, что (p-1)(q-1) = φ(n) – функция Эйлера. Б. Шнайер [4] рекомендует число е выбирать из ряда: 3, 17, 216 + 1.

Наконец расширенный алгоритм Евклида используется для вычисления

третьего компонента ключа: ключа расшифрования, d, такого, чтовыполняется условие:

ed = 1 (mod φ(n)). (8.4)

Другими словами:

d-1 = e(mod φ(n)). (8.5)

Таким образом, сформирован ключ, состоящий из трех чисел, которые, в свою очередь, образуют две вышеупомянутые взаимосвязанные части: открытый (публичный) ключ, (e, n), и тайный ключ, (d, n; на самом деле, как видим, тайным здесь является лишь первое из пары чисел). Примеры генерации ключевой информации, как и ее использования, можно найти, например, в [2].

Использование ключа. Для зашифрования/расшифрования используется ключ получателя: отправитель шифрует сообщение открытым ключом, а получатель расшифровывает шифртекст своим тайным ключом.

Зашифрование. Если шифруется сообщение М, состоящее из r блоков: m1,m2 , …, mi,…,mr, то шифртекст С будет состоять из такого же числа (r) блоков, представляемых числами:

ci = (mi)e mod n. (5.3)

Расшифрование. Для расшифрования каждого зашифрованного блока производится вычисление вида:

mi = (ci) d mod n. (5.4)

Разработаны несколько версий стандарта рассматриваемого алгоритма. Среди прочего, в этих документах обсуждаются размеры безопасного ключа. Доступна одна из последних версий стандарта RSA: RFC 3447. Размер ключа в алгоритме RSA связан с размером модуля,n. Два числа p и q, произведение которых равно n, должны иметь приблизительно одинаковую длину, поскольку в этом случае найти сомножители (факторы) сложнее, чем в случае, когда длина чисел значительно различается. Например, если предполагается использовать 768-битный модуль, то каждое число должно иметь длину приблизительно 384 бита. В 1999 году 512-битный ключ был вскрыт за семь месяцев [4]. Это означает, что 512-битные ключи уже не обеспечивают достаточную криптостойкость. Сейчас в критических системах применяются ключи длиной 1024 и 2048 бит. Ссылочные представления этих чисел в десятичной системе счисления даны в [8].

Алгоритм Эль-Гамаля был предложен Эль-Гамалем (T. El-Gamal) в 1985 г. Он может быть использован для решения трех основных криптографических задач: для зашифрования/расшифрования данных, для формирования цифровой подписи и для согласования общего ключа. Кроме того, возможны модификации алгоритма для схем проверки пароля, доказательства идентичности сообщения и другие варианты.

Как подчеркивалось выше, безопасность алгоритма Эль-Гамаля, как и безопасность алгоритма Диффи-Хеллмана, основана на трудности вычисления дискретных логарифмов. Алгоритм Эль-Гамаля фактически использует схему Диффи-Хеллмана, чтобы сформировать общий секретный ключ для абонентов, передающих друг другу сообщение, и затем сообщение шифруется путем умножения его на этот ключ. И в случае шифрования, и в случае формирования цифровой подписи каждому пользователю необходимо сгенерировать пару ключей.

Рассматриваемый алгоритм отличается от алгоритма RSA несколькими

параметрами и особенностями:

1) генерацией ключевой информации и числом компонент, составляющих ключ;

2) каждому блоку (символу) открытого сообщения в шифртексте на основе алгоритма Эль-Гамаля соответствуют 2 блока (в RSA – один-один);

3) в алгоритме Эль-Гамаля при зашифровании используется число (обозначим его k), которое практически никак не связано с ключевой информацией получателя и которое принимает (по определению) различные значения при зашифровании различных блоков сообщения.

Генерация ключевой информации. Выбирается простое число, р. Выбирается число (g, g < p), являющееся первообразным корнем числа р – очень важный элемент с точки зрения безопасности алгоритма (см. ниже).

Далее выбирается число х (х < p) и вычисляется последний компонент

ключевой информации:

y =gх mod р. (8.6)

Владельцу сформированной ключевой информации, состоящей из 4 чисел, может посылаться некоторый шифртекст, созданный с использованием

открытого ключа получателя: p, g, y. Расшифрование шифртекста получатель

производит своим тайным ключом: p, g, х.

Как видим, на самом деле тайным является лишь одно число (как и в RSA): х.

Определение 2. Первообразный корень (primary (residual ) root ) по модулю р является таким числом, что его степени (gi, 1 ≤i≤p-1 ) дают все возможные по модулю р вычеты (остатки), которые взаимно просты с p.

Понятно, что для больших значений р количество всех неповторяющихся остатков (р – 1) будет также большим. А поскольку в равнении (8.6) мы используем модуль р большого простого числа и находим первообразным корень от р, который имеет важное свойство: при использовании разных степеней (аi = ах) решение будет равномерно распределяться от 0 до р – 1, то нахождение криптоаналитиком нужного х чрезвычайно затруднено. В этом заключается односторонность функции, задаваемой (8.6). И на этом основывается криптостойкость шифра Эль-Гамаля.

Для схемы вероятностного шифрования само сообщение и ключ не определяют шифртекст однозначно. Зашифрование сообщения. Как ранее, предположим, что сообщение М ={mi}, где – mi – i-й блок сообщения.

Зашифрование отправителем (каждого отдельного блоками исходного

сообщения) предусматривает использование, как это особо подчеркивалось выше, некоторого случайного числа k (1 < k <p – 1). В силу использования случайной величины k шифр Эль-Гамаля называют также шифром многозначной замены, а также схемой вероятностного шифрования.

При примерно одинаковой размерности ключей рассмотренные алгоритмы обеспечивают примерно одинаковый уровень криптостойкости.

1. **Практическая часть**

В рамках лабораторной работы был разработан код, имитирующий алгоритм RSA:

Изначально есть один файл в котором ранится информацию для шифрования. После ввода значений p, q и проверки их на простоту, из файла считывается информация и шифруется при помощи алгоритма RSA (рисунок 2.1). В цикле из файла считываются все символы и переводятся в верхний регистр, т.к. в эталонной таблице алфавита (которая была задана в начале программы) прописаны символы русского алфавита, цифры и знаки пунктуации, пробел. После находится n как произведение p и q, так же находится функция Эйлера, которая в коде обозначена переменной m. Она так же находится как как произведение p и q, но предварительно и от p и от q отнимается единица.

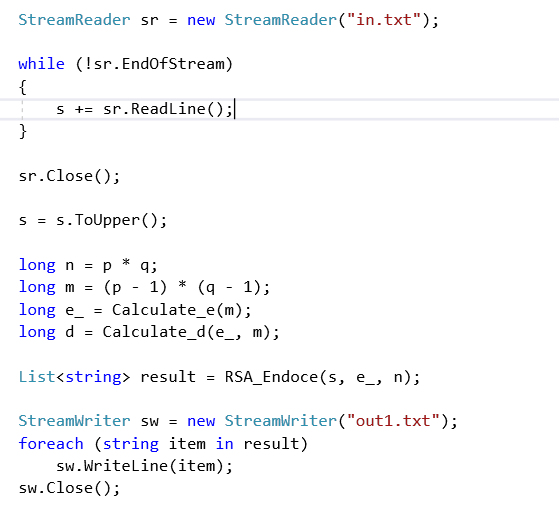


Рисунок 2.1 – Алгоритм шифрования сообщения из файла

При шифровании используются значение таких переменных как e и d, значения которых вычислялись при помощи соответствующих функций. (рисунок 2.2). Эти переменные должны удовлетворять условиям, согласно алгоритмам.

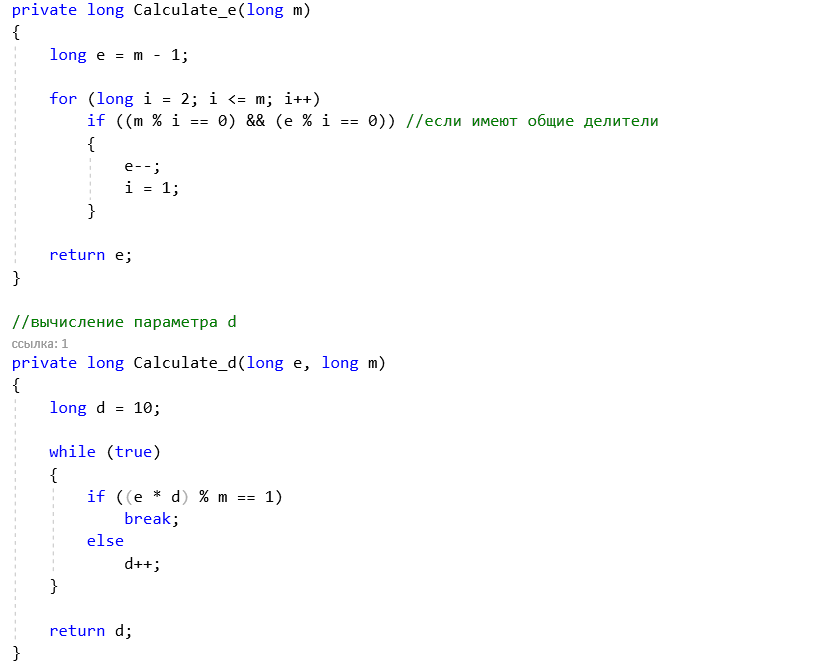


Рисунок 2.2 – Функции вычисления ключей e и d

Реализация алгоритма RSA для шифрования и дешифрования сообщения представлена на рисунках 2.3. и 2.4 соответственно.

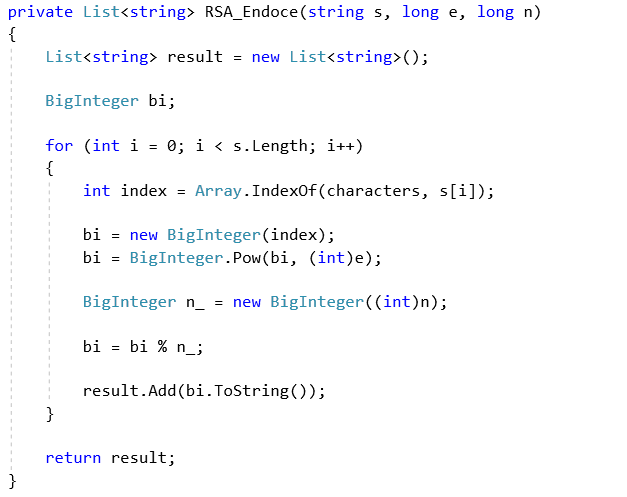


Рисунок 2.3 – Алгоритм шифрования RSA

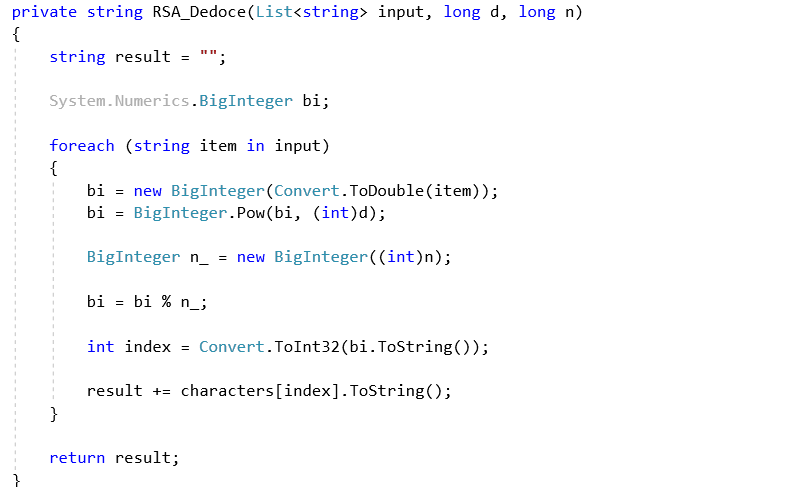
****

Рисунок 2.4 – Алгоритм расшифрования RSA

Результаты шифрования и дешифрования записывались в соответствующие тексовые файлы.

Алгоритм Эль-Гамаля так же был реализован программно. Поскольку для шифрования нам необходимы математические операции, не предусмотренные в стандартном пакете математических функций, необходимо определить нахождение чисел обратных по модулю от произведения чисел и от возведенного в какую-либо степень. Реализация данных функций представлена на рисунках 2.5 и 2.6.

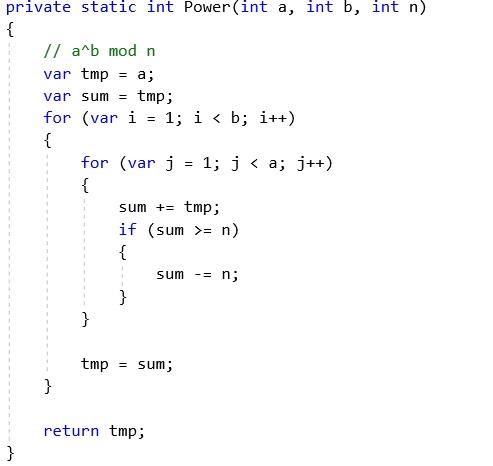


Рисунок 2.5 – Определение функции Power

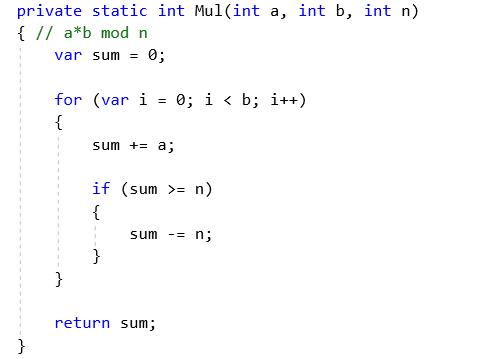


Рисунок 2.6 – Определение функции Mul

Для шифрования сообщения использовались функции, представленные на рисунках 2.7 и 2.8.

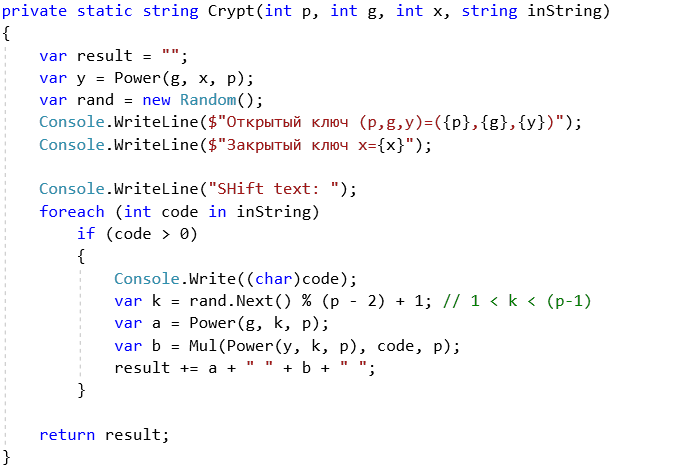


Рисунок 2.7 – Определение функции шифрования сообщения

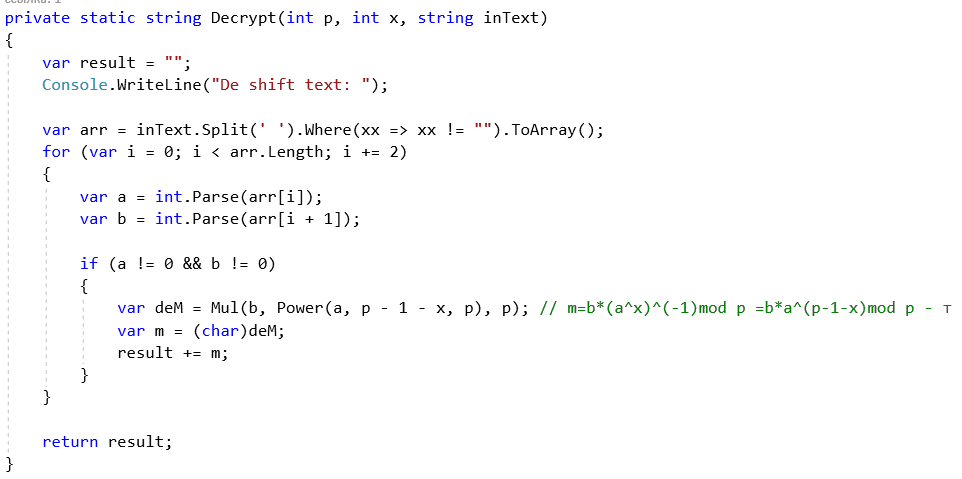


Рисунок 2.8 – Определение функции расшифрования сообщения

Данные ключей и сообщение передаются в функцию путем преобразований, основанных на функциях, определенных в начале происходит процесс шифрования каждого символа. В результате написанного программного приложения получается консольный вывод, представленный на рисунке 2.9.

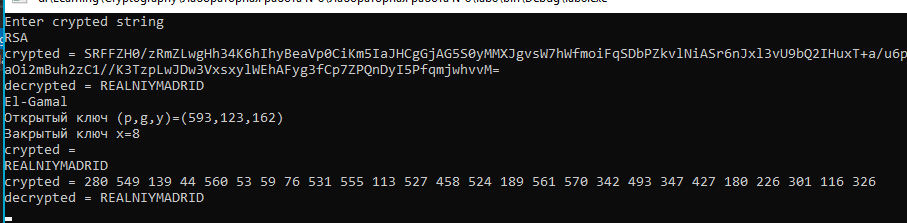


Рисунок 2.9 – Скриншот выполнения программы

**Вывод**

В ходе лабораторной работы были получены знания принципе работы ассиметричных шифров. Изучены алгоритмы RSA и Эль-Гамаля. Так же был реализован программный код, которые динамически осуществляет процесс шифрования и дешифрования информации, полученной из текстового файла при помощи алгоритма RSA и Эль-Гамаля.