01.00.00 Physico-mathematical sciences

01.00.00 Физико-математические науки

UDC 004.056.062

RSA Algorithm. Features of the C # Object Programming Implementation

Elena V. Staver

Belarusian State University, Republic of Belarus
4, Nezavisimosti Ave., 220030, Minsk
Master of Sci. Sciences
E-mail: mindi1987@mail.ru

Abstract. Public-key algorithms depend on the encryption key and the decoding key, connected with the first one. For data public key encryption, the text is divided into blocks, each of which is represented as a number. To decrypt the message a secret key is used.

Keywords: algorithm; encryption; decryption; key.

Введение. Алгоритмы шифрования с открытым ключом зависят от одного ключа для шифрования и другого, связанного с первым, ключа для дешифрования. Эти алгоритмы имеют следующую важную особенность – с точки зрения вычислений нереально определить ключ дешифрования, зная только используемый криптографический алгоритм и ключ шифрования. Для шифрования данных по открытому ключу, текст разбивается на блоки, каждый из которых представляется в виде числа. Для дешифрования сообщения, используя секретный ключ [1].

Материалы и методы. Перед тем как обратиться к криптосистеме с открытым ключом, каждая сторона должна генерировать пару ключей. Это означает выполнение следующих задач:

• определение двух простых чисел p и q;

• выбор одного из чисел e или d и вычисление второго.

Сначала рассмотрим процедуру выбора p и q. Ввиду того что значение n = pq будет известно любому потенциальному противнику, то для того, чтобы не допустить возможности нахождения p и q с помощью простого перебора вариантов, эти простые числа должны быть выбраны из достаточно большого множества (т.е. p и q должны быть большими числами) [1]. В то же время метод нахождения больших простых чисел должен быть практически эффективным.

Для проверки того, что числа простые, существует целый ряд тестов. Почти все такие тесты носят вероятностный характер. Это значит, что тест определит только, что данное целое число, вероятно, простое. Несмотря на отсутствие полной уверенности, такие тесты могут выполняться так, чтобы обеспечить уверенность с вероятностью, как угодно близкой к 1.

Обсуждение. Процедура проверки простоты данного целого числа n заключается в выполнении ряда вычислений, в которых используется n и некоторое случайно выбранное целое число a. Если n не выдерживает тестирования, то n простым не является [1].

Если n выдерживает одно тестирование, то n может оказаться простым, а может и не быть простым. Если же n успешно проходит целый ряд таких "испытаний" с различными случайно выбранными значениями a, это дает нам большую степень уверенности в том, что n на самом деле является простым числом [2].

Результаты. RSA – алгоритм, основанный на использовании разных ключей при шифровании и дешифровании сообщения. На передающей стороне сообщение шифруется с помощью открытого ключа, на приёмной стороне сообщение дешифруется с помощью секретного ключа [2]. Таким образом, устраняется проблема, связанная с обменом ключами между абонентами.

Алгоритм RSA работает следующим образом [2]:

1. Случайным образом выбираются 2 секретных простых числа;
2. Вычисляется произведение

$$n = pq \quad (1)$$

3. Вычисляется значение

$$\Phi = (p-1)(q-1) \quad (2)$$

4. Выбирается открытый K_O и секретный K_C ключи (являются взаимно простыми с Φ и удовлетворяют условию

$$K_O \cdot K_C \bmod \Phi = 1 \quad (3)$$

Для шифрования данных Ко необходимо:

разбить исходный текст на блоки, каждый из которых может быть представлен в виде числа $M(i) = 2,\ldots,n-1$.

Зашифровать эту последовательность чисел по формуле:

$$C(i) = \left(M(i)^{K_O}\right) \bmod n \quad (4)$$

Для расшифровки данных секретным ключом K_C используется следующая формула:

$$M(i) = \left(C(i)^{K_C}\right) \bmod n \quad (5)$$

В результате получим множество чисел $M(i)$, представляющих собой исходный текст.

Защищённость алгоритма RSA основана на сложности разложения числа на простые множители [3].

Для вычисления произведения $n = pq$ и значения $\Phi = (p-1)(q-1)$, в языке C# существует возможность использования следующего программного кода:

```
private void cmdRun2_Click(object sender, System.EventArgs e)
                                        {
                                    int p = Convert.ToInt32(this.txtA.Text);
                                    int q = Convert.ToInt32(this.txtB.Text);
                                        if (p == q)
                                        {
MessageBox.Show("Числа p и q не д.б. равны!!!", ".:Error message:.");
                                    txtSecret.Clear();
                                        return;
                                        }
                            this.txtNOD.Text = NOD(p, q).ToString();
                                this.txtMult.Text = EvklidMult(p, q).ToString();

                                    this.lbn.Text = "N=p*q = ";
                                this.lbF.Text = "Ф= (p-1)(q-1) = ";

                                E = Convert.ToInt32(this.txte.Text);
                                        n = p * q;
                                int F = (p - 1) * (q - 1);
```

Рис. 1. Вычисление произведения $n = pq$

Для генерации открытого K_O и секретного K_C ключей по формуле $K_O \cdot K_C \bmod \Phi = 1$, используется фрагмент кода [3]:

```
public int EvklidMult(int a, int b)
{
int rez = 0;
 int Q = 0;
int x1 = 1, x2 = 0, x3 = b;
int y1 = 0, y2 = 1, y3 = a;
int t1 = 0, t2 = 0, t3 = 0;
while((true) && (b*x1 + a*x2 == x3) &&  (b*t1 + a*t2 == t3))
{
if(y3 == 0)
{
rez = x3;
{
```

Рис. 2. Генерация ключей

Функция шифрования-дешифрования

```
public String ShifrDesh(String str, int key)
{
String result = "";
for (int j = 0; j < str.Length; j++)
{
char symbol = str[j];
int b = (int)symbol;
//if (n == 35)
symbol = (char)(symbol - 1070);
long ressymbol = symbol % n;
```

Рис. 3. Шифрование-дешифрование

Выводы. В результате проведения исследований была разработана программа шифрования и дешифрования с помощью алгоритма RSA.

RSA представляет собой систему асимметричного шифрования, то есть в данном случае шифрование и дешифрование проводится с использованием разных ключей. На передающей стороне сообщение шифруется открытым ключом, на принимающей стороне сообщение дешифруется секретным ключом. Это позволяет избежать проблем, связанных с обменом ключами между абонентами.

Примечания:

1 Шнайер Б. Прикладная криптография. М.: Триумф, 2002. 816 с.

2 Вилльям С. Криптография и защита сетей: принципы и практика. СПб.: Вильямс, 2001. 672 с.

3 Романец Ю.В., Тимофеев П.А., Шаньгин В.Ф. Защита информации в компьютерных системах и сетях. М.: Радио и связь, 2001. 376 с.

УДК 004.056.062

Алгоритм RSA. Особенности реализации на языке объектно-ориентированного программирования C#

Елена Владимировна Ставер

Белорусский государственный университет, Беларусь
220030, Минск, пр.Независимости 4
Магистр, преподаватель
mindi1987@mail.ru

Аннотация. Алгоритмы шифрования с открытым ключом зависят от одного ключа для шифрования и другого, связанного с первым, ключа для дешифрования. Для шифрования данных по открытому ключу, текст разбивается на блоки, каждый из которых представляется в виде числа. Для дешифрования сообщения, используя секретный ключ.

Ключевые слова: алгоритм; шифрование; дешифрование; ключ.