УДК 004

Реализация алгоритма побитового шифрования на языке C++

Ленкин Алексей Викторович
Приамурский государственный университет имени Шолом-Алейхема.
Студент

Аннотация
В данной статье рассмотрен созданный алгоритм шифрования. Описана реализация данного алгоритма с помощью языка программирования C++.
Ключевые слова: C++, шифрование, побитовое шифрования

Implementation of the algorithm of bitwise encryption in C + + language

Lenkin Aleksei Viktorovich
Sholom-Aleichem Priamursky State University
Student

Abstract
This article discusses the generated encryption algorithm. The implementation of this algorithm with the help of C++ programming language is described.
Keywords: C++, encryption, bitwise encryption

Научный руководитель:
Лучанинов Дмитрий Васильевич
Приамурский государственный университет имени Шолом-Алейхема
старший преподаватель кафедры информационных систем, математики и методик преподавания

Шифрование информации на сегодняшний день является главным способом защиты любой информации. На данный момент существует и разрабатывается огромное число методов кодирования информации.

Но иногда имеющиеся виды шифрования не подходят некоторым пользователям или не внушают доверия, в таком случае любой разработчик может создать свой собственный метод шифрования.

Целью исследования является описать один из алгоритмов шифрования перестановкой и создать его программную реализацию на языке C++.

Исследованиями в данной теме занимались следующие авторы. Г.Ю.Протодьяконова, П.И.Стручков, В.Ю.Табырынов, А.И.Соловьев описали «Подходы к разработке криптографического интегрированного алгоритма шифрования» [1]. «Методика разработки асимметричных алгоритмов шифрования данных» была исследована в работе А.А.Назиров [2]. А.С. Поляков организовал «Простой способ разработки «легких» алгоритмов шифрования» [3]. Д.В. Гусева и Д.П. Горейда описали

«Разработка метода шифрования для беспроводных сетей связи» [4]. «Разработка алгоритма шифрования на основе тригонометрической функции двух аргументов» была сделана Д.В.Гусева, С.В.Яковлев [5].

Разработанное в рамках исследования шифрование было создано на основе полиномиального кода. Но существенно отличается от него алгоритмом и реализацией.

Данное шифрование происходит следующим образом, соблюдая следующий алгоритм:

1. Берётся кодируемое сообщение и сообщается его 8-битный ключ шифрования, указанный в двоичном формате (пример «10101010»).
2. Кодируемое сообщение переводится в двоичную систему по таблице ASCII.
3. Кодируемое сообщение умножается на ключ шифрования в двоичной системе счисления.
4. Сообщение зашифровано.

Приведём пример:

1. Возьмём слово «Test» и ключ, к примеру «10101010».
2. Переведём сообщение в двоичную систему, получив «01010100011001010111001101110100».
3. Умножим двоичное сообщение на ключ, вычисления в столбик представлены ниже:

```
                  0 1 0 1 0 1 0 0 0 1 1 0 0 1 0 1 0 1 1 1 0 0 1 1 0 1 1 1 0 1 0 0
                                                          0 1 0 1 0 1 0 1
= = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
                  0 1 0 1 0 1 0 0 0 1 1 0 0 1 0 1 0 1 1 1 0 0 1 1 0 1 1 1 0 1 0 0
                0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
              0 1 0 1 0 1 0 0 0 1 1 0 0 1 0 1 0 1 1 1 0 0 1 1 0 1 1 1 0 1 0 0
            0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
          0 1 0 1 0 1 0 0 0 1 1 0 0 1 0 1 0 1 1 1 0 0 1 1 0 1 1 1 0 1 0 0
        0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
      0 1 0 1 0 1 0 0 0 1 1 0 0 1 0 1 0 1 1 1 0 0 1 1 0 1 1 1 0 1 0 0
    0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
  0 1 0 1 0 1 0 0 0 1 1 0 0 1 0 1 0 1 1 1 0 0 1 1 0 1 1 1 0 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
= = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
0 0 1 1 1 0 0 0 0 0 0 0 1 0 1 1 0 1 0 1 1 1 1 0 1 0 1 0 1 0 1 1 0 0 0 0 1 0 0
```

4. Полученный шифр будет таким «00111000000001011010111101010110000100».

Данный метод кодирования является довольно простым, но его расшифровка может занять довольно долгое время, если не будет известен ключ. Декодирование будет обратным процессом – делением зашифрованного сообщения на ключ.

Реализуем данный алгоритм на языке C++. Разработанная программа состоит из нескольких функций «binary» и связка «polinom» и «plusim».

Функция binary переводит сообщение в двоичную систему счисления. Связка «polinom» и «plusim», совершает собственно само кодирование, умножая по алгоритму столбика. Листинг представлен ниже:

```
#include <iostream>
#include <string>
#include <windows.h>
#include <algorithm>

using namespace std;

string binary(string str)
{
     string sum,ans,temp;
     int a;
     for (int i=0;i<str.length();i++)
     {
          a=(int)str[i];
          while (a!=0)
          {
               sum+=char(a%2)+48;
               a/=2;
          }
          while (sum.length()!=8) sum+='0';
          reverse(sum.begin(),sum.end());
          ans+=sum;
          sum.clear();
     }
    return ans;
}

string plusim (string a, string b)
{
     string d;
     int c=0;
     int g=a.length();
     int t=b.length();

     if (a.length()<b.length())
     {
          for (int i=0;i<abs(g-t);i++) a.insert(a.begin(),'0');
     }
     else
     {
          for (int i=0;i<abs(g-t);i++) b.insert(b.begin(),'0');
     }

     for (int i=a.length()-1;i>-1;i--)
     {
          c+=(static_cast<int>(a[i])-48)+(static_cast<int>(b[i])-
48);
          if (c>=2)
```

```
          {
               d+="0";
               c=1;
          }
          else if (c<=1)
          {
               d+=static_cast<char>(c+48);
               c=0;
          }
          else
          {
               d+="1";
               c=1;
          }
}
     if (c!=0) d+=static_cast<char>(c+48);
     reverse(d.begin(),d.end());
     return d;
}

string polinom(string a,string b)
{
     int c=0;
     string d,s;
     for (int j=a.length()-1;j>-1;j--)
     {
          for (int i=b.length()-1;i>-1;i--)
          {
               d+=static_cast<char>((static_cast<int>(a[j])-
48)*(static_cast<int>(b[i])-48)+48);
          }
          reverse(d.begin(),d.end());
          for (int k=1;k<a.length()-j;k++) d+="0";
          s=plusim(d,s);
          d.clear();
     }
     return s;
}

int main()
{
     setlocale(LC_ALL, "Russian");
     SetConsoleCP(1251);
     SetConsoleOutputCP(1251);
     string s,key;
     cout<<"Введите кодируемое сообщение:"<<endl;
     getline(s,cin);
     cout<<"Введите 8-битный ключ:"<<endl;
     cin>>key;
     cout<<"Двоичное представление
сообщения:"<<endl<<binary(s)<<endl;
     cout<<"Зашифрованное
сообщение"<<endl<<polinom(binary(s),key)<<endl;
```

```
}
```

Попробуем с помощью программы зашифровать фразу «Test word». Результат на рисунке 1.

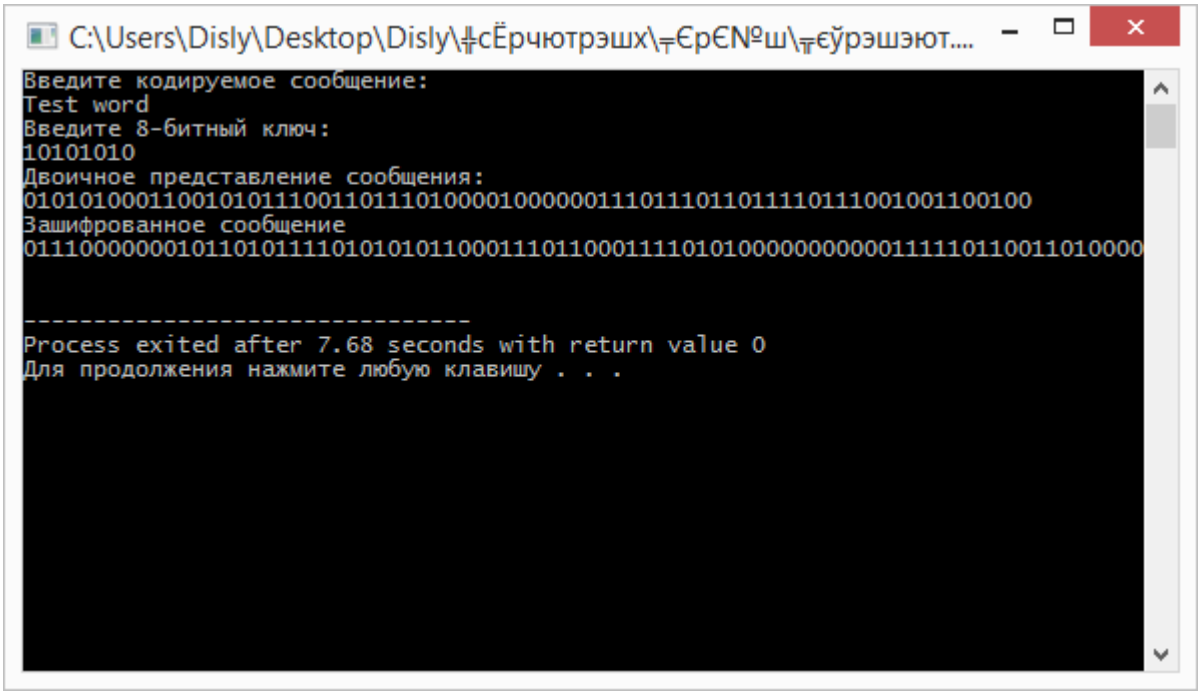

Рисунок 1. Программа для побитового шифрования данных

Если попробовать посчитать вручную, то полученный результат будет таким же.

Таким образом, можно сказать, что разработанный метод шифрования является эффективным способом зашифровать сообщение и довольно криптостойким, так как подбор ключа займёт большой промежуток времени. Но существенным минусом является то, что при шифровании увеличивается размер сообщения.

Библиографический список

1. Протодьяконова Г.Ю., Стручков П.И., Табырынов В.Ю., Соловьев А.И. Подходы к разработке криптографического интегрированного алгоритма шифрования // Academy. 2017. Т. 2. № 6 (21). С. 14-15.
2. Назиров А.А. Методика разработки асимметричных алгоритмов шифрования данных // Актуальные научные исследования в современном мире. 2016. № 7-1 (15). С. 35-37.
3. Поляков А.С. Простой способ разработки «легких» алгоритмов шифрования // Доклады Белорусского государственного университета информатики и радиоэлектроники. 2017. № 2 (104). С. 11-16.
4. Гусева Д.В., Горейда Д.П. Разработка метода шифрования для беспроводных сетей связи // В сборнике: Студенческая наука для развития информационного общества сборник материалов IV Всероссийской научно-технической конференции: в 2-х томах. 2016. С. 61-64.

5. Гусева Д.В., Яковлев С.В. Разработка алгоритма шифрования на основе тригонометрической функции двух аргументов // Труды Северо-Кавказского филиала Московского технического университета связи и информатики. 2016. Т. 1. № 9. С. 311-317.