广西科学院学报
Journal of Guangxi Academy of Sciences

2007,23(4):246~248,251
Vol. 23,No. 4 November 2007

基于C++的加密解密置换算法的实现

The Implementation of a Transposition Algorithm of the Encryption and Decryption Based on C++

黄珍生
HUANG Zhen-sheng

(广西民族大学,广西南宁 530006)
(Guangxi University for Nationalities,Nanning,Guangxi,530006,China)

摘要:分析数据加密技术中的置换算法及其加密解密思想,给出加密解密置换算法的程序设计方法与具体实现步骤,并提供一个完整的加密解密算法的C++源程序。该算法的实现可以配合高等学校信息安全或网络安全课程的教学和实验,使学生更好地理解和掌握信息加密与解密原理。

关键词:加密 解密 置换算法
中图法分类号:TP311.1 **文献标识码**:A **文章编号**:1002-7378(2007)04-0246-03

Abstract: This paper analyses the encryption and decryption ideas as well as the transposition algorithm in the data encryption technology, brings forward a program design method and the detailed steps of realizing transposition algorithm ,and offers a complete program based on C++ language. The algorithm can be used in the teaching and testing courses on information security or network security in the universities to help the students know better the principle of encryption and decryption.

Key words: encryption, decryption, transposition algorithm

尽管已有多种技术可用来保障计算机系统和网络的安全性,但近年来,国内外在安全性方面的研究还是主要集中在两个方面,一是以密码学为基础的各种加密措施,如保密密钥算法和公开密钥算法;二是以计算机网络特别是以 Internet 和 Intranet 为对象的通信安全研究。在保障网络通信安全方面所依赖的主要技术,仍然是数据加密技术。数据加密的方法很多,但最基本的加密方法只有两种,即易位法和置换法,其它方法大多是基于这两种方法所形成[1]。为配合信息安全或网络安全课程的教学和实验,使学生更好地理解和掌握信息加密与解密原理,本文通过分析数据加密技术中的置换算法及其加密解密思想,给出了该算法的程序设计方法与具体实现步骤,并提供一个完整的加密解密的C++源程序。

1 问题分析

置换法是按照一定的规则,用一个字符去置换(替代)另一个字符来形成密文[2]。最早由朱叶斯·凯撒提出的置换算法非常简单,它通过将字母 a、b、c、…、x、y、z 循环右移三位后,形成 d、e、f、…、a、b、c 字符序列,再利用移位后的序列中的字母去分别置换未移位序列中对应位置的字母,即利用 d 置换 a,用 e 置换 b 等等[3]。凯撒算法的推广是移动 K 位[2]。但是,单纯移动 K 位的置换算法还是很容易被破译,比较好的置换算法是进行映像[4],如表 1 所示,将上行的源字符毫无规则地映像到下行的密码字符,利用表 1 上行源码字符一一对应下行密码字符,可将明文译为密文,如将明文 information 加密后变为密文 ofygkdqzogf,将明文 computer 加密后变为密文 egdhxztk;同样,利用表 1 下行密码字符一一对应上行源码字符,可将密文译为明文,如将密文 lnlztd 解码为明文 system 等等。

置换法实质上与发电报很类似,发电报时把发

收稿日期:2007-09-16
作者简介:黄珍生(1965-),男,讲师,主要从事计算机软件研究。

报人所写的每一个汉字变换成一个四位数字码发出,而接收电报时又把收到的每一个四位数字码译成汉字送给收报人。所不同的是电报码为公开的,而置换法的转换码为秘密的。解决问题的关键是建立一个转换码的密码本文件(即通常所说的密码本),它的每一个分量存放两个字符,一个是源码字符,另一个是密码字符。如表1就可以作为某密码本文件的一部分。

表1 密码本示例

源码	…	a	b	c	d	u	v	w	x	y	z	…	1	2	3	4	5	6	…
密码	…	q	w	e	r	x	c	v	b	n	m	…	0	9	+	−	*	/	…

2 算法设计与实现步骤

2.1 算法设计思想

利用计算机程序实现加密解密置换算法,首先建立好密码本文件,建立密码本文件的方法与建立一般结构性文件的方法样,这里假设密码字典文件已经建立好(提供的源程序中也有建立密码本文件的功能),文件名为"dictionary.dat",密码本文件保存在可移动磁盘(如优盘)便于个人保存,加密解密时可将其插入磁盘驱动器(如U盘驱动器)中。其次,采用两个字符数组,一个用于存放密码本的上行(源码字符),另一个用于存放密码本的下行(密码字符),读入数组之后,立即取出移动磁盘自行保管,以防失密,退出程序后临时存放密码字符的数组也不复存在了。最后再实现加密解密过程,加密过程就是扫描源文件(明文),对于它的每个字符都到字典中去找出与之相同的源码字符,而后取出对应的密码字符送入密码文件(密文);解密过程就是扫描密码文件(密文),对它的每一个字符都到字典中找与之相同的密码字符,而后取出对应的源码字符送入源文件形成明文。

2.2 算法模块划分与实现步骤

算法划分为主模块,读字典模块,加密模块和解密模块这四大程序模块来实现。

2.2.1 主模块的实现步骤

(1)说明接受密码本文件上行(源码字符)、下行(密码字符)的两个数组,说明文件变量f1、f2及dictionf,分别对应被译制文件(明文)、译制结果文件(密文)及密码本文件。

(2)假定明文存放在文件F:\123.dat中,并调用读密码本文件模块。

(3)列出主菜单,按功能序号选择功能。

1——加密,2——解密,3——建立密码本,0——退出。

(4)接收用户选择,并根据选择以不同参数调用功能模块。

(5)结束。

2.2.2 读字典模块的实现步骤

(1)提示用户将含密码本文件的U盘插入U盘驱动器中。

(2)建立内部文件变量dictionf与外部文件dictionary.dat的联系。

(3)打开密码本文件用于读,将字典文件的上、下行分别读入数组1与数组2中。

(4)关闭文件,提示用户取出密码本文件所在的磁盘,自己妥善保管,以防失窃失密。

(5)结束。

2.2.3 加密模块的实现步骤

(1)设置两个数组参数a1、a2,第一个对应翻译前的字符码,第二个对应翻译后的字符码。

(2)建立f1与被译文件(F:\123.dat)的联系及f2与结果文件(F:\temp.dat)的联系。

(3)打开f1用于读,打开f2用于写。

(4)扫描f1,当f1没有取完字符时,则循环执行:

{ ①从f1中取一个字符,存于ch中;

②在第一个数组中找到一个与ch匹配的元素:a1[j];

③根据下标j在a2数组中找到a2[j](即翻译后的字符),将a2[j]存送结果文件f2中

}

(5)关闭文件f1和f2。

(6)结束。

2.2.4 解密模块的实现步骤

解密模块的实现步骤与加密模块基本相同,略。

3 算法的C++实现

以下是完整的加密解密置换算法的C++语言源程序,本程序在VC++6.0[5]上调试、编译通过。

```
#include<iostream.h>
#include<stdlib.h>
#include<string.h>
#include<fstream.h>
#include<ctype.h>
#define size 256//密码本中包涵的元素个数
struct dict {
```

```
    char source;//密码本文件源码字符(上行)
    char secret;//密码本文件密码字符(下行)
        }dict1;
char diction1[size],diction2[size];//用于存放密码本的两个数组
char ch;//当前要加密的字符
int j;//临时变量
char filename[20];//存放等待加密的外部文件名
void readdictionary()  {//读密码本模块
    cout<<"请将含密码本的U盘插入驱动器中,按回车键继续"<<endl;
    cin.get();//起暂停作用,按键继续
    ifstream dictionf("F:\dictionary.dat",ios::in|ios::nocreate);
     //打开密码本文件用于输入
    dictionf.seekg(0);
    dict dict2;
    int m=sizeof(dict2);
    int j=0;
    while ( dictionf.read((char *)&dict2,m))
    {  j=j+1;
       diction1[j]=dict2.source;//数组diction1存密码本的源码字符
       diction2[j]=dict2.secret;//数组diction2存密码本的密码字符
    };
    dictionf.close( );
    cout<<"读取密码本完成,请将U盘取出妥为保管,按回车键继续!"<<endl;  cin.get();
};
void encryption( char a1[size],char a2[size]) //加密模块
{  ifstream f1("F:\123.dat");//打开要加密的文件,用于输入
    if (f1.fail()) {
       cerr<<"要加密的文件不存在"<<endl;
       exit(1);
    }
    ofstream f2("F:\temp.dat");//打开存放结果的文件,用于输出
    if (f2.fail()) {
       cerr<<"存放结果的文件不存在"<<endl;
       exit(1);
    }
    ch=f1.get();
    int j;
    while (ch! =EOF)
    {  j=1;
       while (ch!=a1[j]) j=j+1;//要加密的字符与密码本源码匹配
       ch=a2[j];//如匹配,换码,将明文换成密文
       f2.put(ch);
    ch=f1.get();//取下一个字符
    };
    cout<<"文件加密成功,按回车确定"<<endl;
    cin.get();
};
void decryption( char a1[size],char a2[size]) //解密模块
{  ifstream f1("F:\temp.dat");//打开要解密的文件,用于输入
    if (f1.fail()) {
       cerr<<"要解密的文件不存在"<<endl;
       exit(1);
    }
    ofstream f2("F:\123.dat");//打开存放结果的文件,用于输出
    if (f2.fail()) {
       cerr<<"存放结果的文件不存在"<<endl;
       exit(1);
    }
    ch=f1.get();
    int j;
    while (ch! =EOF)
    {  j=1;
       while (ch!=a1[j]) j=j+1;//要解密的字符与密码本密码字符匹配
       ch=a2[j];//如匹配,换码,将密文换成明文
       f2.put(ch);
    ch=f1.get();
    };
    cout<<"文件解密成功,按回车确定"<<endl;
    cin.get();
};
void mimazidian()
{  fstream fout("F:\dictionary.dat",ios::out|ios::trunc|ios::binary);  if (! fout) {
       cerr<<"文件 mimaben.dat 不存在!"<<endl;
```

(下转第251页)

理、有效.

参考文献:

[1] 于卫平. 基于离散 Hopfield 神经网络的数据加密算法研究[J]. 湖南理工学院学报:自然科学版,2007(3):24-25,94.

[2] 张健,周洁敏. 基于 TMS320C54x DSP 的加密算法设计[J]. 电子元器件应用,2006(3):67-69.

[3] 陈永强. 一种基于混沌加密的自适应图像水印方法[J]. 计算机应用,2007(10):2453-2455.

[4] 潘天骥. 论述中国剩余定理的形成及其影响[J]. 九江师专学报:自然科学版,1987,5(1):1-4.

[5] 曹珍富. 公钥密码学[M]. 哈尔滨:黑龙江出版社,1993:2.

(责任编辑:尹 闯 邓大玉)

(上接第 248 页)

```
        exit(1);
    }
    dict x;
    cout<<"请输入明文字符与密文字符的对应表形成密码本文件,按 Ctrl+z 键结束:"<<endl;
    while (cin>>x.source) {
        cin>>x.secret;
        fout.write((char *)&x,sizeof(x));
    }
    fout.close();
    cout<<"输入结束并已在 F 盘形成密码本文件。"<<endl;
};
void main() {//主程序模块
    readdictionary( );//调用读密码本文件模块
    cout<<"请按下列菜单选择功能:"<<endl;
    cout<<"1——加密    2——解密    3——建立密码本    0—退出"<<endl;
    cout<<"请选择(0————3):";
int i;
    cin>>i;
    switch(i) {
    case 1:
        encryption(diction1,diction2);//加密
        break;
    case 2:
        decryption(diction2,diction1);//解密
        break;
    case 3:
        mimazidian();//生成密码本文件
        break;
    case 0:
        return;
    }
}
```

4 结束语

本文用 C++语言给出了一种加密解密置换算法的具体实现,探讨了西文字符的加密解密问题,但本算法可以扩展到中文甚至其他文字的加密与解密,但是这方面的加密解密问题稍微复杂一些,有待进一步探讨。

参考文献:

[1] WADE TRAPPE, LAWRENCE C WASHINGTON. 密码学概论[M]. 北京:人民邮电出版社,2004.

[2] 范辉,谢青松. 操作系统原理与实训教程[M]. 第 2 版. 北京:高等教育出版社,2006.

[3] 汤子瀛. 计算机操作系统[M]. 西安:西安电子科技大学出版社,2002.

[4] 甘仞初. 信息系统原理与应用[M]. 北京:高等教育出版社,2004.

[5] RICHARCL C LEINECKER, TOM ARCHER. Visual C++6 宝典[M]. 北京:电子工业出版社,2001.

(责任编辑:韦廷宗)