

杭州电子科技大学

实验报告

课程名称：密码学课程设计 姓名：苏展 学号： 1827139

实验地点：科技馆 620

实验时间：2020-3-3

一、实验名称： RC4 密码实验

二、实验要求：

- 1、了解流密码的起源与涵义。
- 2、掌握 RC4 密码的加解密原理。
- 3、用 Visual C++实现 RC4 密码程序并输出结果。

三、实验内容：

1、1949 年，Shannon 发表了《保密系统的通信理论》，奠定了现代密码学的基础。Shannon 还证明了一次一密的密码体制是绝对安全的。一次一密指的是密钥取成与明文等长的 0-1 随机序列，加密时把密钥和明文逐位做异或，解密时就将密钥与密文做异或。可是，一次一密体制要求的是密钥长度等于明文长度，这在实践中是不实用的。而流密码可以看成是为了实用化而模仿一次一密的一类体制。简单地说，流密码利用比明文短得多的密钥生成伪随机的密钥流，再将该密钥流当成“一次一密”中的密钥进行加密与解密。

2、RC4 是密码学家 Ronald Rivest 在 1987 设计的一种流密码，现在在网络通信中的应用十分广泛。它的描述如下：

明文 $m = m_1m_2 \cdots m_n$ 是字符序列， $m_i \in [0,255]$ 。

密钥 $K = K_1K_2 \cdots K_s$ 是字符序列， $K_i \in [0,255]$ ， $s \in [5,16]$ 称为 KeySize。

密钥流 $k = k_1k_2 \cdots k_n$ 是字符序列， $k_i \in [0,255]$ 。

密文 $c = c_1c_2 \cdots c_n$ 也是字符序列， $c_i \in [0,255]$ 。

S 盒 是一个长度为 256 的字符数组 S[256]，它是 $[0,255] \rightarrow [0,255]$ 的双射。

1) S 盒初始化

```
unsigned char S[256];
```

```
for i from 0 to 255
```

```
S[i] = i;
```

2) 利用密钥 **K** 打乱 **S** 盒

```
j = 0;
```

```
for i from 0 to 255
```

```
{
```

```
    j = ( j + S[i] + K[i mod Keysize] ) mod 256;
```

```
    交换 S[i], S[j];
```

```
}
```

3) 利用 **S** 盒生成伪随机密钥流 **k**

设明文序列长度为 n ,

```
i = 0;
```

```
j = 0;
```

```
for t from 0 to n-1
```

```
{
```

```
    i = ( i + 1 ) mod 256;
```

```
    j = ( j + S[i] ) mod 256;
```

```
    交换 S[i], S[j];
```

```
    k[t] = S[ ( S[i] + S[j] ) mod 256];
```

```
}
```

4) 加密

将密钥流与明文逐位作异或，得到密文， \wedge 即为逐位异或运算符

```
for t from 0 to n-1
```

```
c[t] = m[t]  $\wedge$  k[t];
```

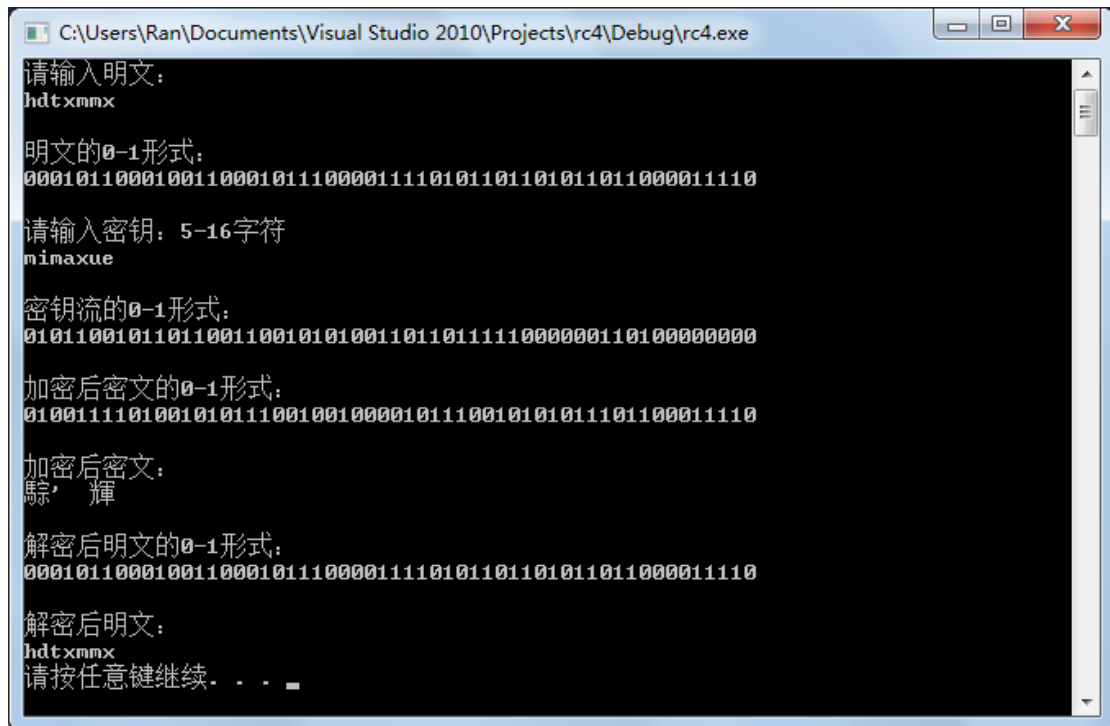
5) 解密

将密钥流与密文逐位作异或，得到解密后的明文

```
for t from 0 to n-1
```

```
m1[t] = c[t]  $\wedge$  k[t];
```

3、使用 Visual C++编写程序，实现 RC4 密码及输出界面，可参考如下：



```
C:\Users\Ran\Documents\Visual Studio 2010\Projects\rc4\Debug\rc4.exe
请输入明文:
hdtxmmx

明文的0-1形式:
000101100010011000101110000111101011011011011000011110

请输入密钥: 5-16字符
mimaxue

密钥流的0-1形式:
01011001011011001100101010011011011111000000110100000000

加密后密文的0-1形式:
01001111010010101110010010000101110010101011101100011110

加密后密文:
鯨 輝

解密后明文的0-1形式:
000101100010011000101110000111101011011011011000011110

解密后明文:
hdtxmmx
请按任意键继续. . .
```

主要步骤:

- 1) 新建一个空项目，取名为 RC4。
- 2) 在左边的解决方案资源管理器中添加 cpp 文件，取名为 RC4.cpp。
- 3) 在 RC4.cpp 中先写入

```
#include<stdlib.h>

#include<iostream>          //用于使用 cin、cout 输入输出函数

using namespace std;

void main()

{

    //在此编写 RC4 密码程序

}
```
- 4) 提示输入明文字符序列
- 5) 输出明文的 0-1 形式
- 6) 提示输入密钥字符序列
- 7) 生成与明文等长的密钥流并输出 0-1 形式
- 8) 加密：将明文和密钥流逐位异或，输出密文 0-1 形式
- 9) 输出密文

10) 解密：将密文和密钥流逐位异或，输出解密后明文的 0-1 形式

11) 输出解密后明文

提示点：

1) 字符数组需定义成无符号形式，例 `unsigned char S[256]`;

`unsigned char` 类型范围：0~255

`char` 类型范围：-128~127

RC4 密码程序代码如下：

```
#!/usr/bin/python
# -*- coding: UTF-8 -*-

from PyQt5 import QtCore, QtGui, QtWidgets
import sys
from Ui_RC4 import Ui_MainWindow

class Min(QtWidgets.QMainWindow,Ui_MainWindow):
    def __init__(self,parent=None): #ui 部分
        super().__init__()
        self.setupUi(self)
        self.jm.clicked.connect(self.func1)
        self.jm2.clicked.connect(self.func2)
        self.qk.clicked.connect(self.clear)
        self.m1=[]
        self.c=[]
        self.kstr=[]
    def start(self):
        j=0
        #读取密钥
        self.key=list(self.my.toPlainText())
        #读取明文
        self.m=list(self.mw.toPlainText())
        #初始化S 盒
        self.S=[i for i in range(256)]
        #利用密钥打乱S 盒
        for i in range(256):
            j=(j+self.S[i]+ord(self.key[i%len(self.key)]))%256
            self.S[i],self.S[j]=self.S[j],self.S[i]
        #生成密钥流
        i=0
        j=0
```

```

        for k in range(len(self.m)):
            i=(i+1)%256
            j=(j+self.S[i])%256
            self.S[i],self.S[j]=self.S[j],self.S[i]
            self.kstr.append(self.S[(self.S[i]+self.S[j])%256])

def func1(self):
    self.start()
    #加密
    for i in range(len(self.m)):
        self.c.append(chr(int(self.kstr[i]^ord(self.m[i])))
    #显示密文
    self.mw2.setPlainText(''.join(self.c))
    #显示密钥流
    kstr=[str(i) for i in self.kstr]
    self.myl.setPlainText(''.join(kstr))

def func2(self):
    self.start()
    #解密
    for i in range(len(self.m)):
        self.m1.append(chr(int(self.kstr[i]^ord(self.c[i])))
    #显示明文
    self.jmw.setPlainText(''.join(self.m1))
    #显示密钥流
    kstr=[str(i) for i in self.kstr]
    self.myl.setPlainText(''.join(kstr))

def clear(self):
    self.mw.setPlainText('')
    self.mw2.setPlainText('')
    self.my.setPlainText('')
    self.myl.setPlainText('')
    self.jmw.setPlainText('')
    self.m1=[]
    self.c=[]
    self.kstr=[]

if __name__ == '__main__':
    app = QtWidgets.QApplication(sys.argv)
    ui=Min()
    ui.show()
    sys.exit(app.exec_())

```

输出结果截屏如下：

RC4

明文

密码学真是太厉害了太厉害了太厉害了

密钥

厉害

密钥流

6520898196140162234724318623919112822416116686652089819614016
2234724318623919112822416116686

密文

宇磴嫪矛囟妈匪宜礼蚣卦幡丌姊匪燿仝

解密文

密码学真是太厉害了太厉害了太厉害了

加密

解密

清空