

Министерство образования и науки РФ  
Федеральное государственное бюджетное образовательное учреждение  
высшего профессионального образования

Тульский государственный университет

КАФЕДРА ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

**МЕТОДЫ ПРОЕКТИРОВАНИЯ ПО ИНФОРМАЦИОННОЙ  
БЕЗОПАСНОСТИ НА ОСНОВЕ ТЕХНОЛОГИИ ВИЗУАЛЬНОГО  
ПРОГРАММИРОВАНИЯ**

Лабораторная работа № 7  
по курсу «Методы и средства защиты компьютерной информации»

Выполнил:	студент группы 220601	<hr/>	Белым А.А.
		(подпись)	
Проверил:	д. т. н., проф. каф. ВТ	<hr/>	Данилкин Ф.А.
		(подпись)	

Тула 2013

## Цель работы

Освоение теоретических положений и получение практических навыков разработки ПО на основе технологии визуального программирования, позволяющего защищать авторские права.

## Задание

Разработать ПО, позволяющее модифицировать первые 512 байт файла с расширением exe любого файла. Модификация заключается в побитном сложении по модулю 2 первых 512 байт файла с расширением exe с заданными произвольным образом 512 байтами.

## Текст программы

Далее представлен текст программы на языке C++, реализующей модификацию исполняемых файлов.

Файлmainwindow.h:

```
#ifndef MAINWINDOW_H
#define MAINWINDOW_H

#include <QMainWindow>

namespace Ui {
class MainWindow;
}

class MainWindow : public QMainWindow
{
    Q_OBJECT

public:
    explicit MainWindow(QWidget *parent = 0);
    ~MainWindow();

private slots:
    void on_pushButton_clicked();

    void on_pushButton_3_clicked();

    void on_pushButton_5_clicked();

private:
    QString infile,outfile;
    unsigned char mod[512];
    Ui::MainWindow *ui;
};

#endif // MAINWINDOW_H
```

Файлmainwindow.cpp:

```

#include "mainwindow.h"
#include "ui_mainwindow.h"
#include <QFileDialog>
#include <QMessageBox>
#include <cstdio>
#include <fstream>
#include <vector>
#include <iomanip>
#include "des.h"
MainWindow::MainWindow(QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::MainWindow)
{
    ui->setupUi(this);
}

MainWindow::~MainWindow()
{
    delete ui;
}

void MainWindow::on_pushButton_clicked()
{
    infile=QFileDialog::getOpenFileName(this);
    ui->label->setText(infile);
}

QString rand512() {
    QString t,t1;
    for(int i=0;i<512;i++)
        t+=t1.sprintf("%02x",rand()%256);
    return t;
}

bool parseHex(QString &str,unsigned char v[512]){
    if (str.length()!=1024)
        return false;
    bool ok;
    for(int i=0;i<512;i++){
        v[i]=str.mid(2*i,2).toInt(&ok,16);
        if(!ok)
            return false;
    }
    return true;
}

void MainWindow::on_pushButton_3_clicked()
{
    ui->plainTextEdit->setPlainText(rand512());
}

void filemod(std::iostream& fs,unsigned char mod[512]){
    std::vector<unsigned char> buf;
    buf.insert(buf.begin(),mod,mod+512);
    unsigned char t;int i,j;
    fs>>std::noskipws;
    for(i=0;i<512&&fs>>t;i++){
        buf[i]^=t;
    }
    std::cout<<i<<" bytes read"<<std::endl;
    fs.clear();fs.seekp(0);
    for(j=0;j<i;j++){
        fs<<unsigned char(buf[j]);
    }
    std::cout<<j<<" bytes wrote"<<std::endl;
}

```

```

void MainWindow::on_pushButton_5_clicked()
{
    if(!parseHex(ui->plainTextEdit->toPlainText(),mod))
        QMessageBox::warning(this,"","Неправильный ключ!");

    std::fstream
    ifile(infile.toUtf8().constData(),std::ios::in|std::ios::out|std::ios::binary);
    filemod(ifile,mod);
};

```

## Тестовый пример

На рисунке 1 представлен пример исходного файла перед модификацией, открытого в hex-редакторе.

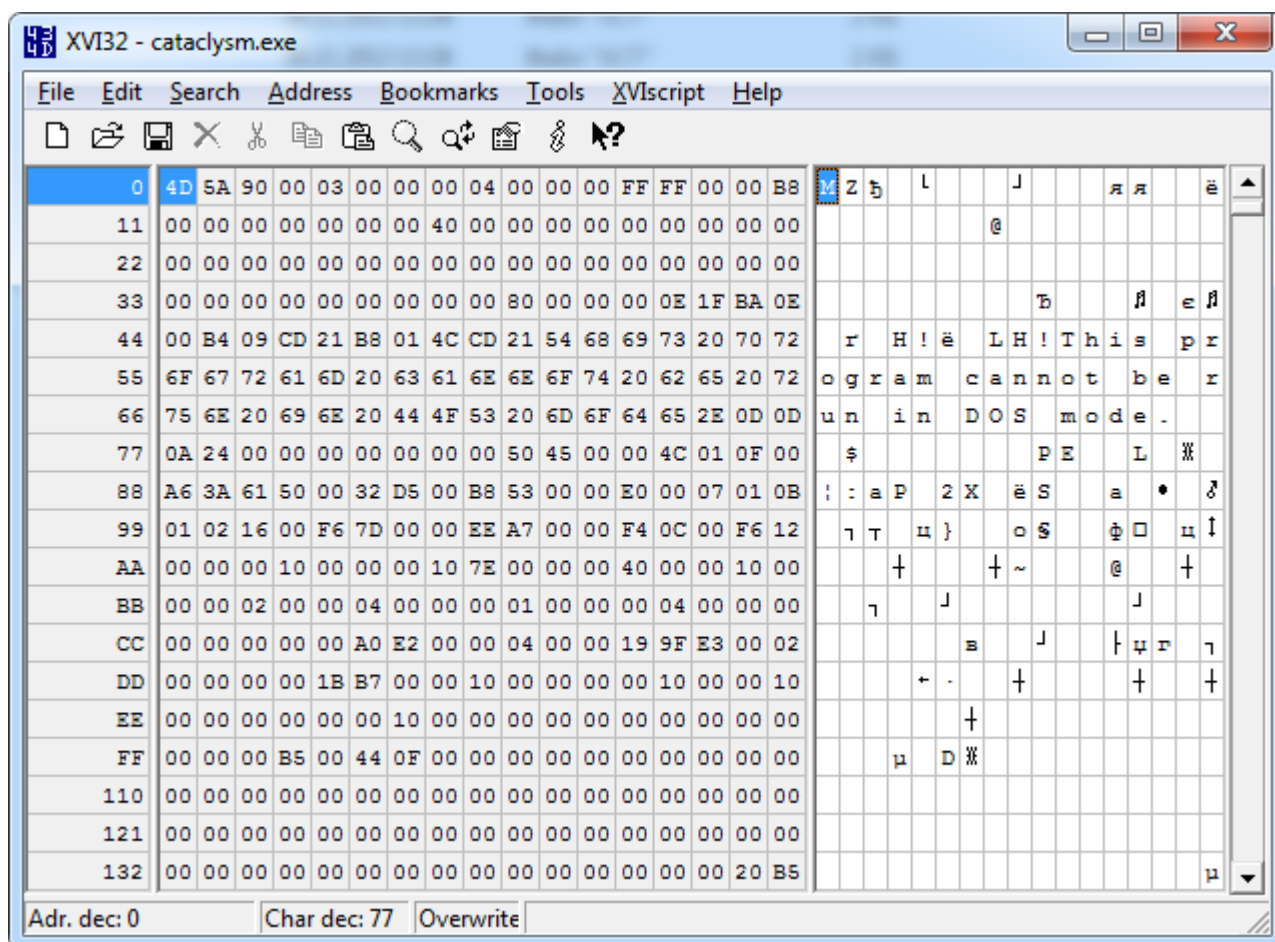


Рисунок 1 – Пример исходного файла

На рисунке 2 представлен пример работы программы при модификации исходного файла.

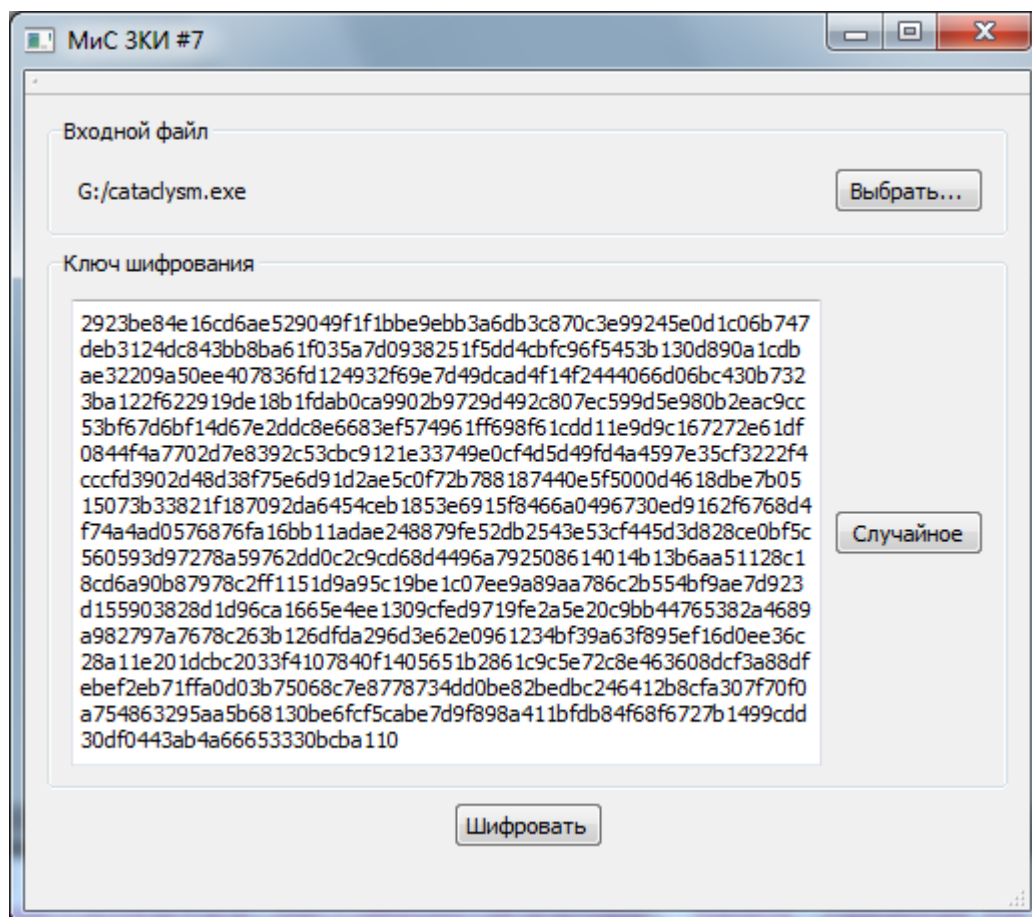


Рисунок 2 – Пример работы программы

На рисунке 3 представлен пример файла после модификации.

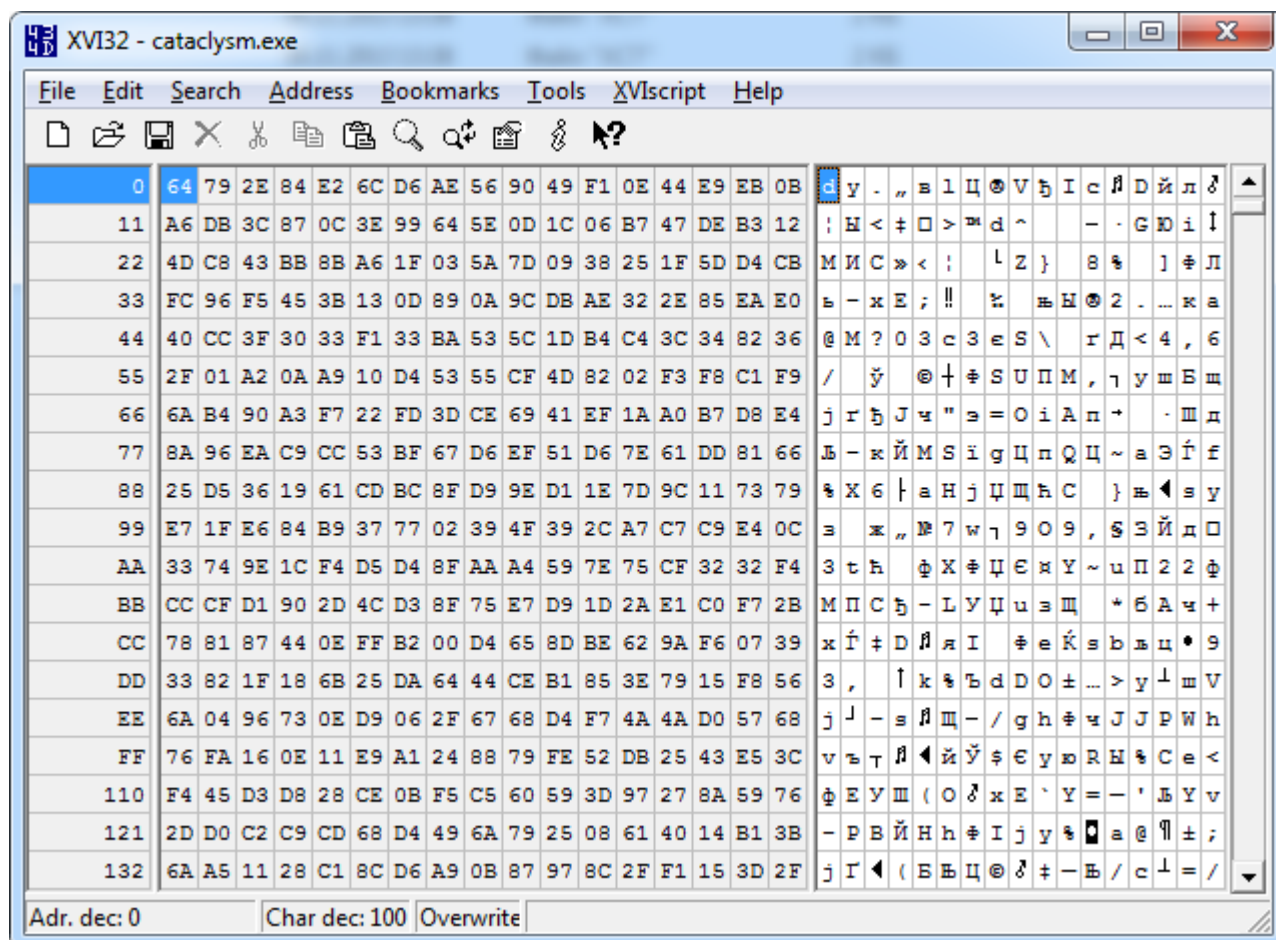


Рисунок 3 – Пример модифицированного файла

## Вывод

В данной работе рассмотрены простейшие методы защиты исполняемых файлов. Для предотвращения кражи или декомпиляции программы могут использоваться простые арифметические преобразования, как, например, сложение по модулю 2. В данной работе разработана программа, позволяющая сложить по модулю 2 начальные байты файла с заданным 512 байтовым ключом.