­Министерство высшего и профессионального образования РФ

Государственное образовательное учреждение

высшего профессионального образования

Ульяновский государственный технический университет

Кафедра «Вычислительная техника»

Дисциплина "МиСЗКИ"

Лабораторная работа №5

Изучение принципов защиты от самораспространяющихся программ.

Выполнил:

студент группы БЭВМд-41

Ключников Дмитрий

Проверил:

Мартынов А. И.

Ульяновск 2012

***Задание:***

Изучить принципы защиты от самораспространяющихся программ и разработать программу-доктора, которая выполняет следующие действия:

1. Ищет «зараженные» файлы в указанном каталоге (подкаталогах) и выводит статистику по анализу содержимого файлов. Под «зараженным» понимается только та программа, которая содержит код самораспространения, реализованный в лабораторной работе №5 согласно варианту. То есть программа-доктор не должна производить лечение «чужих» программ.
2. При обнаружении зараженного файла, вынимает из него «тело вируса» и восстанавливает работоспособность программы.

**Алгоритм работы антивируса:**

1. Ищет на диске подходящий COM - файл.

2. Проверяем его размер на ограничение.

3. Проверяем, заражен ли он.

4. Если да, то лечим его:

1. Восстанавливаем заголовок исходя из данных, содержащихся в вирусе для корректного запуска зараженной программы
2. Удаляем тело вируса
3. Обновляем данные в заголовке.
4. Обновляем файл

**Код программы:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.IO;

namespace Scaner

{

class Program

{

/// <summary>

/// Литера, говорящая что данный файл заражен

/// </summary>

private static string VIRUS\_LABEL = "K";

/// <summary>

/// Количество байт тела вируса

/// </summary>

private static int VIRUS\_LENGTH = 753;

/// <summary>

/// Количество байт с конца файла до оригинальных байтов перехода

/// </summary>

private static int startGenuine = 362;

/// <summary>

/// Некоторый кусок тела вируса

/// </summary>

private static byte[] vir = new byte[] { 68, 109, 105, 116, 114, 121, 32, 75, 108, 121, 117, 99, 104, 110, 105, 107, 111, 118, 32, 66, 95, 69, 86, 77, 100, 45, 52, 49, 10, 13, 86, 97, 114, 105, 97, 110, 116, 32, 35, 49, 58, 32, 10, 13, 86, 105, 114, 117, 115, 32, 105, 110, 102, 101, 99, 116, 115, 32, 67, 79, 77, 45, 102, 105, 108, 101, 115, 32, 115, 105, 122, 101, 32, 108, 101, 115, 115, 32, 50, 53, 107, 98, 32, 97, 110, 100, 32, 97, 116, 116, 114, 105, 98, 117, 116, 101, 115, 58, 32, 78, 111, 114, 109, 97, 108, 44, 32, 65, 114, 99, 104, 105, 118, 101, 44, 32, 72, 105, 100, 100, 101, 110, 46, 10, 13, 10, 13, 36, 10, 13, 68, 111, 32, 121, 111, 117, 32, 119, 105, 115, 104, 32, 116, 111, 32, 115, 116, 97, 114, 116, 32, 118, 105, 114, 117, 115, 63, 32, 40, 89, 47, 78, 41, 10, 13, 36, 73, 110, 112, 117, 116, 32, 99, 111, 117, 110, 116, 32, 102, 105, 108, 101, 115, 58, 32, 36, 10, 13, 70, 105, 108, 101, 32, 73, 110, 102, 101, 99, 116, 58, 32, 36, 0, 1, 46, 75 };

static void Main(string[] args)

{

Console.WriteLine("Начать поиск зараженных файлов в данной папке, рекурсивно? Нажмите 'Enter'");

var listFilesInfect = new List<string>();

if (Console.ReadKey().Key == ConsoleKey.Enter)

{ // Да, начинаем поиск

var files = Directory.GetFiles(Directory.GetCurrentDirectory(), "\*.com", SearchOption.AllDirectories);

foreach (var fileName in files.Where(Isinfected))

{//Если находим инфицированный файл, отображаем и сохраняем путь

Console.WriteLine("{0} заражен!", fileName.Replace(Directory.GetCurrentDirectory() + "\\", ""));

listFilesInfect.Add(fileName);

}

if (listFilesInfect.Count > 0)

{

Console.WriteLine("Лечить найденные файлы? Нажмите 'Enter'");

if (Console.ReadKey().Key == ConsoleKey.Enter)

{

// Лечим

foreach (var fileName in listFilesInfect.Where(CureFile))

{ // Если вылечили - говорим об этом

Console.WriteLine("{0} вылечен!", fileName);

}

}

}

}

Console.WriteLine("Все!");

Console.ReadKey();

}

/// <summary>

/// Лечение файла от вируса

/// </summary>

/// <param name="fileName"></param>

/// <returns></returns>

private static bool CureFile(string fileName)

{

try

{

var bytes = File.ReadAllBytes(fileName);

var orgBytes = bytes.Skip(bytes.Length - startGenuine).Take(3).ToArray();

// заменяем первые 3 бита на исходные

for (int i = 0; i < 3; i++)

bytes[i] = orgBytes[i];

// Убираем тело вируса

bytes = bytes.Take(bytes.Length - VIRUS\_LENGTH).ToArray();

// Сохраняем d файл

File.WriteAllBytes(fileName, bytes);

return true;

}

catch (Exception)

{

return false;

}

}

static bool Isinfected(string fileName)

{

var stream = File.OpenRead(fileName);

stream.Position = stream.Length - 1;

var byteEnd = stream.ReadByte();

if (VIRUS\_LABEL != Encoding.ASCII.GetString(new[] { (byte)byteEnd }))

return false; // Если нету в конце литиры - значит не инфицирован

stream.Position = stream.Length - 206;

for (int i = 0; i < vir.Length - 5; i++)

{

// Сравнивание некоторого куска файла на предмет тела вируса

var byteV = stream.ReadByte();

if (byteV == vir[i]) continue; // Если нашли несовпадение - значит это другой файл

stream.Close();

return false;

}

stream.Close();

return true;

}

}

}