**1. ЦЕЛИ И ЗАДАЧИ ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ**

Целью выполнения данной курсовой работы является закрепление и углубление теоретических знаний в области защиты информации в компьютерных системах при использовании антивирусных пакетов и межсетевых экранов. А также для приобретения навыков разработки программных средств, реализующих некоторые функциональные элементы для анализа работоспособности средств защиты компьютеров.

Задачи для выполнения курсовой работы:

1. Изучить теоретически основы построения систем защиты компьютерных систем от вредоносных программ и систем применяемых для защиты межсетевых взаимодействий.
2. Уметь применять теоретические знания о системах защиты компьютерных систем от вирусов и о системах, применяемых для защиты межсетевых взаимодействий.
3. Изучить и освоить программные способы контроля работоспособности антивирусных пакетов и межсетевых экранов.
4. Выбрать программные средства для реализации средств контроля работоспособности антивирусных пакетов и межсетевых экранов.
5. Разработать алгоритмы работы модулей ПО, для контроля работоспособности антивирусного пакета и межсетевого экрана, а также представить их в виде блок-схем.
6. Разработать интерфейс программного обеспечения.
7. Программно реализовать средство контроля работоспособности антивирусного пакета и межсетевого экрана.
8. Выполнить анализ полученных результатов, выявить достоинства и недостатки разработанной системы, и предложить возможные способы дальнейшего развития.
9. Оформить результаты выполнения курсовой работы в виде пояснительной записки в соответствии с предъявляемыми требованиями.

**2. ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ**

В курсовой работе требуется разработать алгоритмы модулей и реализовать на их основе комплексное программное обеспечение для проверки безопасности ПК, предназначенное для контроля работоспособности установленных на ПК антивируса и фаервола.

В соответствии с заданием требуется выполнить анализ работоспособности средств защиты компьютеров от вирусов и межсетевого взаимодействия. Необходимо разработать алгоритм и комплексное программное обеспечение для контроля работоспособности антивируса и межсетевого экрана. Реализовать проверку наличия сетевого подключения; проверку наличия и работоспособности установленного на ПК межсетевого экрана и антивирусного ПО.

Разработанное ПО должно содержать следующие модули:

- Модуль проверки наличия соединения с Интернетом.

- Модуль проверки наличия установленного межсетевого экрана.

- Модуль проверки работоспособности межсетевого экрана.

- Модуль проверки наличия установленного антивируса.

- Модуль проверки работоспособности антивирусного ПО.

- Модуль вывода результатов работы приложения.

- Модуль сохранения результатов проверки.

**3. ВЫПОЛНЕНИЕ КУРСОВОЙ РАБОТЫ**

**3.1. Порядок выполнения курсовой работы**

Курсовая работа (КР) предназначена для самостоятельного практического закрепления материала и выполняется в несколько этапов.

**Подготовительный этап.** Уточнение постановки задачи. Аналитический обзор научно-технической литературы по теме КР.

**Проектный этап.**  Разработка архитектуры приложения и его модулей.

**Реализационный этап.** Программная реализация комплексной системы проверки безопасности персонального компьютера (ПК).

**Оформительский этап.** Оформление пояснительной записки в соответствии с требованиями пунктов 3.2 и 3.3 настоящих методических указаний.

**Заключительный этап.** Защита курсовой работы.

**3.2. Структура и примерное содержание пояснительной записки**

Пояснительная записка к курсовой работе должна включать следующие структурные элементы в указанной ниже последовательности:

* титульный лист (Приложение 1);
* задание на курсовую работу (Приложение 2);
* оглавление;
* введение, в котором раскрываются актуальность и значение темы, формулируется цель работы;
* Разработка программного обеспечения:
  + постановка задачи;
  + разработка и описание алгоритмов комплексной системы проверки безопасности ПК;
  + Программная реализация модулей на основе разработанных алгоритмов;
  + описание программно реализованных функций;
  + результаты применения программы, программного кода и снимки экрана после запуска приложения.
* заключение;
* список использованных источников;
* приложение, содержащее исходные коды разработанной программы с необходимыми комментариями.

**3.3. Требования к оформлению пояснительной записки**

Пояснительная записка к курсовой работе выполняется на стандартных листах бумаги формата А4, сброшюрованных в папку. Текст печатается на одной стороне листа. Поля: левое поле -25 мм, правое, верхнее и нижнее – 15мм. Шрифт: Times New Roman, размер - 14. Межстрочный интервал- 1,5. Страницы текста должны иметь сквозную нумерацию. Первой страницей является титульный лист, на котором номер не проставляется.

**3.4. Разработка алгоритма, интерфейса и программная реализация.**

На первом этапе требуется разработать алгоритмы работы модулей в соответствии, с которым, будет функционировать приложение.

Следующим этапом является разработка интерфейса приложения. Приложение может быть выполнено в виде консольного или иметь оконный интерфейс с расположенными на нем элементами управления. Один из вариантов оконного интерфейса представлен на рисунке 1.

Разработанный интерфейс должен обеспечивать выполнение всех функций в соответствии с составленным алгоритмом.

При программной реализации приложения необходимо, обеспечить выполнения функциональных возможностей заложенных при разработке алгоритма приложения. Рассмотрим в качестве примера программную реализацию, представленную ниже по модульно.

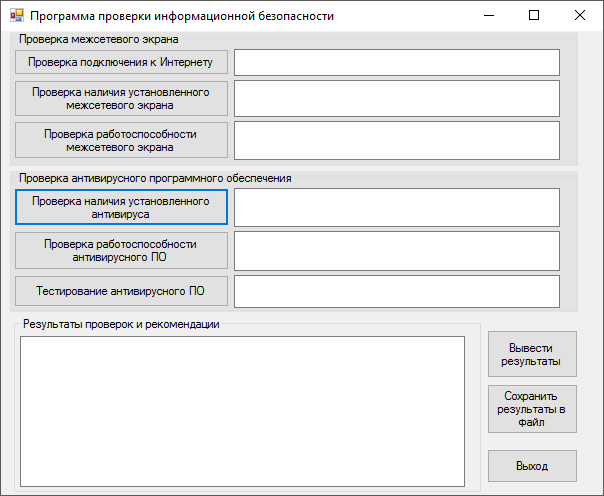


Рисунок 1. Возможный вариант интерфейса приложения.

**3.4.1. Проверка наличия подключения к Интернету.**

Данная проверка должна осуществляться при нажатии на соответствующую кнопку. Проверку можно осуществить с помощью команды "ping". Для этого должен выполняться соответствующий код, подобный следующему:

using System.Net.NetworkInformation;

IPStatus status = IPStatus.Unknown;

try

{

status = new Ping().Send("www……...ru").Status;

}

catch { }

if (status == IPStatus.Success)

{

textBox1.Text = "Данный компьютер подключен к интернету";

}

else

{

textBox1.Text = "Данный компьютер не подключен к интернету";

}

**3.4.2. Проверка наличия установленного межсетевого экрана**

Проверку установленного программного обеспечения можно проверить с помощью нескольких способов, одним из таких способов является проверка наличия папок с установленными файлами или наличия непосредственно самих файлов программного обеспечения. Например, следующим образом:

private void button2\_Click(object sender, EventArgs e)

{

if (!(File.Exists("C:/Program Files/……………...exe")))

{

textBox2.Text = "Фаервол ……….. установлен!";

}

else

{

textBox2.Text = "Фаервол ………………… не установлен!";

}

}

**3.4.3. Проверка работоспособности межсетевого экрана**

Проверку работоспособности межсетевого экрана можно осуществить с помощью обращения из программы к внешнему ресурсу. Если доступ удастся получить и фаервол не сообщит о такой попытке, значит, он не работает или неправильно настроен. Для реализации такого способа, можно использовать программный код представленный ниже.

using System.Net.NetworkInformation;

WebClient Client = new WebClient();

String Response;

try

{

Response = Client.DownloadString("http://www..........com");

}

catch

{

textBox3.Text = "Межсетевой экран функционирует правильно!";

}

if(textBox3.Text=="")

{

textBox3.Text = "Межсетевой экран функционирует неверно, или не функционирует вовсе!";

}

**3.4.4. Проверка наличия установленного антивируса**

Для обеспечения информационной безопасности ПК, на нем должен быть установлен антивирусный пакет. Одним из способов проверки наличия установленного приложения на компьютере является проверка по списку установленных программ в реестре. Примерный код, выполняющий такую операцию, представлен ниже.

private void button6\_Click(object sender, EventArgs e)

{

GetInstalledSoftware();

}

private void GetInstalledSoftware()

{

List<string> items = new List<string>();

using (var hklm = RegistryKey.OpenBaseKey(RegistryHive.LocalMachine, RegistryView.Registry64)) // или 32

using (var rk = hklm.OpenSubKey(@"SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall"))

{

foreach (string skName in rk.GetSubKeyNames())

{

using (RegistryKey sk = rk.OpenSubKey(skName))

{

try

{

if (sk.GetValue("DisplayName") != null)

{

items.Add(sk.GetValue("DisplayName").ToString());

listBox1.Items.Add(new ListViewItem(items.ToArray()));

listBox1.SetSelected(listBox1.Items.Count - 1, true);

if (listBox1.SelectedItem.ToString() == "ListViewItem: {………………………………………..}")

{

textBox4.Text = "Антивирус ………………. установлен в системе";

}

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message, "Exception", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

}

items.Clear();

}

}

}

private string CheckValue(object input)

{

if (input != null)

return input.ToString();

else

return string.Empty;

}

**3.4.5. Проверка работоспособности антивирусного ПО**

При отсутствии в составе антивирусного пакета монитора, работающего постоянно, обеспечивается не полная защита операционной системы от вредоносных программ. Как правило, во всех современных антивирусах присутствует резидентный модуль позволяющий проверять программы и документы в режиме реального времени. Для того, что бы проверить, запущен ли резидентный модуль антивируса в автоматическом режиме, можно использовать программный код, подобный в листинге представленному ниже.

using System.Diagnostics;

private void button5\_Click(object sender, EventArgs e)

{

var runningProcs = from proc in Process.GetProcesses(".") orderby proc.Id select proc;

if (runningProcs.Count(p => p.ProcessName.Contains("s………….")) > 0)

{

textBox5.Text="Антивирус …………….. работает!";

}

else

{

textBox5.Text = "Антивирус …………….. не работает!";

}

}

**3.4.6. Вывод результатов**

По окончании проведения проверок необходимо вывести информацию с их результатами. Результаты выводятся в виде сообщения в отдельном окне или в текстовом поле на основной форме приложения. Результаты можно выводить автоматически после проведения всех проверок или по нажатию на соответствующую кнопку. Рассмотрим способ информирования с помощью текстового поля в приложении после нажатия на кнопку «Вывод результатов». Фрагмент примерного кода представлен ниже. В этом фрагменте представлен код выводящий информацию о проверке наличия соединения с интернетом и наличия установленного на ПК фаервола. Аналогичным образом в это текстовое поле выводятся результаты других проверок.

private void button11\_Click(object sender, EventArgs e)

{

textBox8.Clear();

//Вывод информации о проверки наличия соединения

textBox8.Text = "Результаты проведенного тестирования антивируса и фаервола" + System.Environment.NewLine + System.Environment.NewLine;

if (textBox1.Text == "Да") textBox8.Text = textBox8.Text + "1. Данный компьютер подключен к интернету";

else

{

if (textBox1.Text == "Нет") textBox8.Text = textBox8.Text + "1. Данный компьютер не подключен к интернету";

else

{

textBox8.Text = textBox8.Text + " 1. Тестирование подключения к интернету не выполнялось";

}

}

textBox8.Text = textBox8.Text + System.Environment.NewLine;

//Вывод информации о проверки наличия фаервола

if (textBox2.Text == "Не установлен") textBox8.Text = textBox8.Text + "2. Фаервол ………… не установлен на этом ПК!";

else

{

if (textBox2.Text == "Установлен") textBox8.Text =textBox8.Text + "2. Фаервол ………….. установлен на этом ПК!";

else

{

textBox8.Text = textBox8.Text + "2. Проверка наличия фаервола на данном ПК не выполнялась";

}

}

textBox8.Text = textBox8.Text + System.Environment.NewLine;

. . . . . . . . . . . . .

textBox1.Clear(); textBox2.Clear();

textBox3.Clear(); textBox4.Clear();

textBox5.Clear(); textBox6.Clear();

}

По окончании вывода результатов необходимо очистить поля с 1 по 6 для того что бы избежать ситуации, что при повторной проверке, будут выведены неверные результаты.

**3.4.7. Сохранение результатов в файл.**

Для сохранения результатов выполненных тестов в файл необходимо использовать следующий примерный программный код, представленный ниже. На рисунке 3 представлено окно сохранения результатов.

private void button9\_Click(object sender, EventArgs e)

{

//saveFileDialog1.ShowDialog();

if (saveFileDialog1.ShowDialog() == DialogResult.OK)

{

string filename = saveFileDialog1.FileName;

// сохраняем текст в файл

File.WriteAllText(filename, textBox8.Text);

}

}

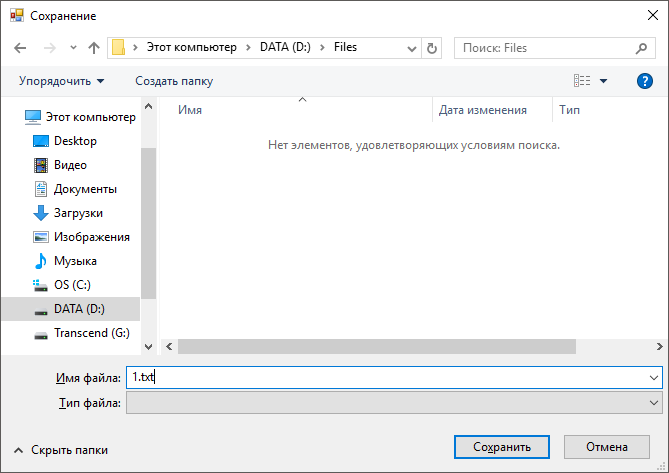


Рис.3. Окно сохранения результатов тестирования

**3.4.8. Выход из программы**

Для более комфортного использования разработанного приложения необходимо предусмотреть кнопку выхода из приложения. Пример кода обработки нажатия кнопки и выхода их приложения представлен ниже.

private void button10\_Click(object sender, EventArgs e)

{

Close();

}

**3.5 Защита курсовой работы**

Защита является заключительным этапом выполнения курсовой работы и проводится не позднее, чем за две недели перед началом сессии.

Пояснительная записка выполненной курсовой работы в электронном виде отправляется на электронный адрес преподавателя для проверки. Преподаватель вносит в текст свои замечания по работе.

После устранения замечаний пояснительная записка выполненной курсовой работы в бумажной форме сдается на проверку руководителю курсовой работы. Руководитель вносит в текст свои замечания по работе, принимает решение о допуске к защите, делая об этом запись на титульном листе, или возвращает работу на доработку с указанием причин.

Защита состоит в демонстрации студентом функционирования разработанной программы и ответах на вопросы преподавателя (по теоретическим аспектам данной темы, алгоритмам и кодам программы в рамках содержания работы). Защита курсовой работы позволяет выявить уровень знаний студента и степень его самостоятельности при выполнении работы.

В результате защиты курсовой работы, согласно Положению о промежуточной аттестации студентов, оценивается дифференцированной отметкой («неудовлетворительно», «удовлетворительно», «хорошо», «отлично»).

**4. СПИСОК РЕКОМЕНДОВАННОЙ ЛИТЕРАТУРЫ**

1. Климентьев К.. Компьютерные вирусы и антивирусы. Взгляд программиста, ДМК Пресс, 656 стр., 2015г.

2. Холмогоров В., PRO Вирусы, ООО «Страта», Санкт-Петербург, 2015г.

3. Черпаков И., Основы программирования. Учебник и практикум, Юрайт, 220 стр., 2016г.

4. Тамара Немцова, Светлана Голова, Алексей Терентьев, Программирование на языке высокого уровня. Программирование на языке С++. Учебное пособие, Форум, 512 стр., 2016г.

5. Н.И. Парфилова, А. Н. Пылькин, Б.Г. Трусов, Программирование. Основы алгоритмизации и программирования. Учебник, 2014г., 240 стр.

**Задание для выполнения курсовой работы по дисциплине**

**«Методы и средства защиты информации в компьютерных сетях»**

1. Выполнить анализ средств защиты компьютеров от вирусов в соответствии вариантом задания.
2. Выполнить анализ средств защиты межсетевого взаимодействия в соответствии вариантом задания.
3. В соответствии вариантом задания разработать алгоритм и комплексное программное обеспечение для контроля работоспособности антивируса и межсетевого экрана. Реализовать в нем:
4. Проверку наличия сетевого подключения;
5. Проверку наличия установленного на ПК и работоспособности межсетевого экрана;
6. Проверку наличия установленного на ПК и работоспособность антивирусного ПО;
7. Вывод результатов на экран;
8. Вывод результатов тестирования в файл;

4. По результатам выполнения работы сделать соответствующие выводы.

Варианты задания:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Язык программирования:   |  |  | | --- | --- | |  |  | | Интерфейс приложения:   |  |  | | --- | --- | | Оконный | Консольный | |  |  | |