# *A subsystem for analyzing data integrity in a computer system*

# *Подсистема анализа целостности данных в компьютерной системе*

***Oleg Choporov, Igor Lvovich, Konstantin Razinkin***

***Олег Чопоров, Игорь Львович, Константин Разинкин***

***Abstract:***

*The paper is devoted to algorithm development and software tools to track changes in software and hardware of the computer. The analysis of modern approaches to the analysis of data integrity. Analyzed the advantages and disadvantages of these approaches. At the moment there are implemented the approaches related to the analysis of the integrity of the computer, but the approach we have developed an integrated manner with hardware and software integrity. To determine the degree of isolation of the computer can be used model speed control. First, check if there are any changes in the BIOS. Then, if everything is in order, it reads the boot sector of a disk or operating system drivers, which in turn also analysed for making unauthorized changes. And, finally, with the help of the operating system runs the driver call control program which ensures that the computer runs only trusted programs. Recommendations for the construction of algorithm for data integrity based on a combination of hardware and software approaches. The analysis of the configuration of the computer for all connected devices and software on the computer. Your computer configuration is copied to all connected devices at a given time to a text file, which is considered as the reference. To determine changes in the structure of the connected devices after some time repeats the sequence of actions to test devices, information is stored in text file comparisons. Consider a reference file and a file compare, conclude whether it has changed the device during the past time interval. Displays a message about adding a new device (or multiple devices) or a message indicating that the system state has not changed. The description of the software product created on the basis of the checksum algorithm and validation*

*algorithm external devices, allowing to track changes in software and hardware of the computer. The advantage of a software product is that it comprehensively analyzes the software and hardware integrity of computer data.*

***Keywords:***

*Information protection, computer system, data integrity.*

***ACM Computing Classification System:***

*Input/output and data communications-Data Communications Devices*
*Input/output and data communications-Input/Output Devices*
*Input/output and data communications-Reliability, Testing, and Fault-Tolerance*

***Abstract:***

*Работа посвящена разработке алгоритма и программного средства, позволяющего отслеживать изменения в программной и аппаратной части компьютера. Проведен анализ современных подходов к анализу целостности данных. Проанализированы достоинства и недостатки этих подходов. На настоящий момент существуют реализованные подходы, связанные с анализом целостности компьютера, но в разрабатываемом нами подходе комплексным образом рассматривалась программная и аппаратная целостность. Для определения степени изолированности компьютера может использоваться модель ступенчатого контроля. Сначала проверяется, нет ли изменений в BIOS. Затем, если все соответствует норме, считывается загрузочный сектор диска или драйвера операционной системы, которые, в свою очередь, также анализируются на предмет внесения в них несанкционированных изменений. И, наконец, с помощью операционной системы запускается драйвер контроля вызовов программ, который следит за тем, чтобы в компьютере запускались только проверенные программы. Даны рекомендации по построению алгоритма целостности данных на основе комбинации аппаратного и программного подходов. Проводится анализ конфигурации компьютера по всем подключенным устройствам и программному обеспечению на данном компьютере. Из конфигурации компьютера копируется информация о всех подключенных устройствах на данный момент времени в текстовый файл, который рассматривается как эталонный. Для выяснения изменения структуры подключенных устройств по истечению некоторого времени повторяется последовательность действий по проверке устройств, информация сохраняется в текстовый файл сравнения. Рассматривают эталонный файл и файл сравнения, делают вывод, изменялись ли устройства за прошедший интервал времени. Выводится сообщение о добавлении нового устройства (или нескольких устройств) или сообщение о том, что состояние системы не изменилось. Дано описание программного продукта, созданного на основе алгоритма контрольной суммы и алгоритма проверки внешних устройств, позволяющего отслеживать изменения в программной и аппаратной части компьютера. Достоинством программного продукта является то, что в нем*

*комплексным образом проводится анализ программной и аппаратной целостности данных компьютера.*

***Keywords:***

*Защита информации, компьютерная система, целостность данных.*

## Введение

Проблемы защиты информации связаны с ростом возможностей современной вычислительной техники. Развитие методик, подходов и способов проведения автоматизации обработки информации, массовость применения персональных компьютеров резко повышают уязвимость информации [1-3].

В качестве основных факторов, которые способствуют увеличению такой уязвимости можно отметить:

- заметный рост объемов информации, которая накапливается, хранится и передается при помощью компьютеров и разных средств автоматизации;

- хранение информации при помощи единых баз данных разного назначения и имеющих разную принадлежность;

- заметное увеличение числа пользователей, которые имеют непосредственный доступ к соответствующим ресурсам вычислительных систем и тем массивам данных, которые там находятся;

- проведение усложнения для режимов работы технических средств в вычислительных системах: активное использование мультипрограммных режимов, и режима разделения времени;

- проведение автоматизации межмашинного обмена информацией, это касается и сетей больших размеров;

В таких условиях возникают возможности для того, чтобы информация была несанкционированным образом использована или модифицирована (есть опасность утечек информации, относящейся к ограниченному пользованию).

Это вызывает особую озабоченность пользователей, в связи с чем защите информации от несанкционированного доступа (чтения) уделяется повышенное внимание.

Специфичная область компьютерных преступлений и то обстоятельство, что их совершают, как правило, люди с незапятнанной репутацией и хорошо владеющие тонкостями информационных технологий затрудняет раскрытие таких преступлений [4-6].

Защита информации неразрывным образом связана с разработкой и использованием информационных систем, это касается как администраторов, так и широкого круга пользователей. В последние годы, в связи с широким распространением и повсеместным применением вычислительной техники, массовостью внедрения компьютеров, резко повысилась уязвимость накапливаемой, хранимой и обрабатываемой в системах информации [7-9].

Приведенные факты показывают, что опасность несанкционированных злоумышленных действий в вычислительных средствах и системах является весьма реальной, и с дальнейшим развитием вычислительной техники угроза повреждения информации, несмотря на все усилия по ее защите, неизменно растет.

Все это обславливает необходимость углубленного анализа опыта защиты информации и комплексной организации методов и механизмов защиты.

Цель работы: Разработка программного средства, с помощью которого можно отслеживать целостность данных среди программного обеспечения и отслеживать изменение аппаратной части компьютера.

Для достижения поставленной цели необходимо было решить ряд задач:

1. Реализовать алгоритм расчета контрольной суммы (CRC32).
2. Разработать алгоритм отслеживания внешних устройств.
3. Разработать программное средство на основе изученных методов.

## 1 Анализ литературных источников

Задача защиты программного обеспечения от внедряемых программных закладок может рассматриваться в принципиально различных вариантах:

– не допустить внедрения программной закладки в компьютерную систему;

– выявить внедренную программную закладку;

– удалить внедренную программную закладку.

При рассмотрении этих вариантов решение задачи защиты от программных закладок сходно с решением проблемы защиты компьютерных систем от вирусов [10, 11]. Как и в случае борьбы с вирусами, задача решается с помощью средств контроля за целостностью запускаемых системных и прикладных программ, а также за целостностью информации, хранимой в компьютерной системе и за критическими для функционирования системы событиями. Однако данные средства действительны только тогда, когда сами они не подвержены влиянию программных закладок, которые могут:

– навязывать конечные результаты контрольных проверок;

– влиять на процесс считывания информации и запуск программ, за которыми осуществляется контроль;

– изменять алгоритмы функционирования средств контроля.

При этом чрезвычайно важно, чтобы включение средств контроля выполнялось до начала воздействия программной закладки либо когда контроль осуществляется только с использованием программ управления, находящихся в ПЗУ компьютерной системы [12-14].

**1.** Защита от внедрения программных закладок. Универсальным средством защиты от внедрения программных закладок является создание изолированного компьютера. Компьютер называется изолированным, если выполнены следующие условия:

– в нем установлена система BIOS, не содержащая программных закладок;

– операционная система проверена на наличие в ней закладок;

– достоверно установлена неизменность BIOS и операционной системы для данного сеанса;

– на компьютере не запускалось и не запускается никаких иных программ, кроме уже прошедших проверку на присутствие в них закладок;

– исключен запуск проверенных программ в каких-либо иных условиях, кроме перечисленных выше, т. е. вне изолированного компьютера.

Для определения степени изолированности компьютера может использоваться модель ступенчатого контроля. Сначала проверяется, нет ли изменений в BIOS. Затем, если все в порядке, считывается загрузочный сектор диска или драйвера операционной системы, которые, в свою очередь, также анализируются на предмет внесения в них несанкционированных изменений. И, наконец, с помощью операционной системы запускается драйвер контроля вызовов программ, который следит за тем, чтобы в компьютере запускались только проверенные программы [15, 16].

Интересный метод борьбы с внедрением программных закладок может быть использован в информационной банковской системе, в которой циркулируют исключительно файлы – документы. Чтобы не допустить проникновения программной закладки через каналы связи, в этой системе не допускается прием никакого исполняемого кода. Для распознавания событий типа «ПОЛУЧЕН ИСПОЛНЯЕМЫЙ КОД» и «ПОЛУЧЕН ФАЙЛ-ДОКУМЕНТ» применяется контроль за наличием в файле запрещенных символов: файл признается содержащим исполняемый код, если в нем присутствуют символы, которые никогда не встречаются в файлах-документах.

**2.** Выявление внедренной программной закладки. Выявление внедренного кода программной закладки заключается в обнаружении признаков его отсутствия в компьютерной системе. Эти признаки можно разделить на следующие два класса:

– качественные и визуальные;

– обнаруживаемые средствами тестирования и диагностики.

К качественным и визуальным признакам относятся ощущения и наблюдения пользователя компьютерной системы, который отмечает определенные отклонения в ее работе (изменяется состав и длины файлов, старые файлы куда–то пропадают, а вместо них появляются новые, программы начинают работать медленнее, или заканчивают свою работу слишком быстро, или вообще перестают запускаться). Несмотря на то, что суждение о наличии признаков этого класса кажется слишком субъективным, тем не менее, они часто свидетельствуют о наличии неполадок в компьютерной системе и, в частности, о необходимости проведения дополнительных проверок присутствия программных закладок.

Признаки, выявляемые с помощью средства тестирования и диагностики, характерны как для программных закладок, так и для компьютерных вирусов. Например, загрузочные закладки успешно обнаруживаются антивирусными программами, которые сигнализируют о наличии подозрительного кода в загрузочном секторе диска. С инициированием статической ошибки на дисках хорошо справляются средства работы с дисковыми накопителями, входящие в Windows. А средства проверки целостности данных на диске, входящие в состав современных антивирусных средств, позволяют успешно выявлять изменения, вносимые в файлы программными закладками. Кроме того, эффективен поиск фрагментов кода программных закладок по характерным для них последовательностям нулей и единиц (сигнатурам), а также разрешение выполнения только программ с известными сигнатурами.

**3.** Удаление внедренной программной закладки. Конкретный способ удаления внедренной программной закладки зависит от метода ее внедрения в компьютерную систему. Если это программно-аппаратная закладка, то следует перепрограммировать ПЗУ компьютера. Если это загрузочная, драйверная, прикладная, замаскированная закладка или закладка – имитатор, то можно заменить их на соответствующую загрузочную запись, драйвер, утилиту или служебную программу, полученную от источника, заслуживающего доверия. Наконец, если это исполняемый программный модуль, то можно попытаться добыть его исходный текст, убрать из него имеющиеся закладки или подозрительные фрагменты, а затем заново откомпилировать.

В разрабатываемом программном продукте комплексным образом проводится анализ программной и аппаратной целостности данных компьютера.

## 2 Алгоритм проверки целостности данных компьютера

Рассмотрим общую схему алгоритма проверки целостности данных компьютера. Она приведена на рисунке 1.

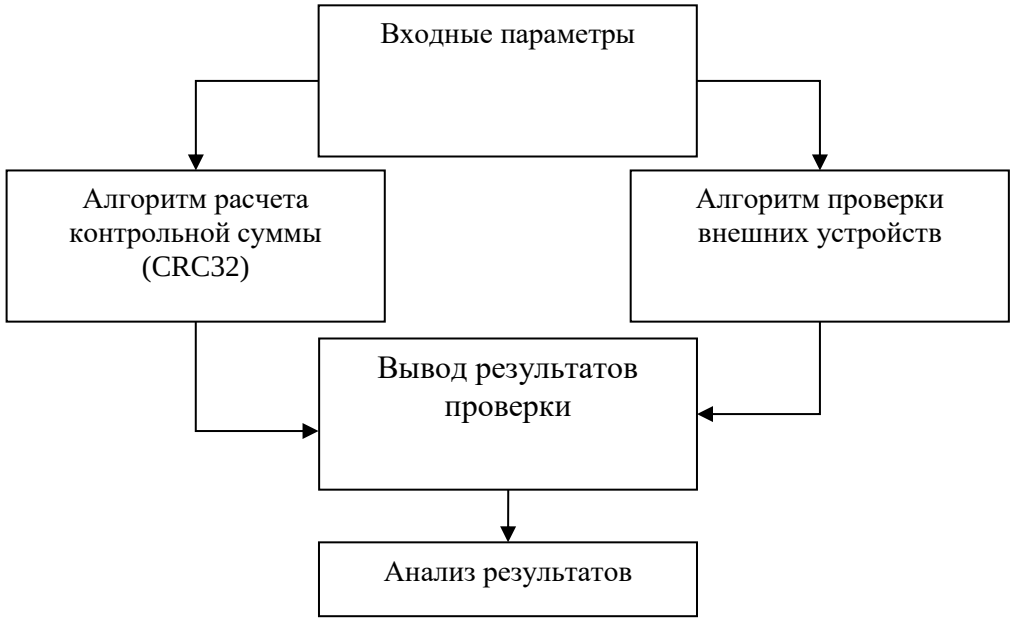


Рисунок 1. Общая схема алгоритма проверки целостности данных

Общая схема, позволяет представить комплексный подход к определению целостности данных в программной и аппаратной части компьютера.

Рассмотрим эту схему поблочно:

1. Входные данные. Здесь под входными данными понимается как любой файл, хранящийся на жестком диске, так и внешние устройства подключенные к компьютеру.

2. Алгоритм расчета CRC32. Данный алгоритм позволяет отследить целостность данных в программной части персонального компьютера, а также производить их проверку через интервалы времени выбираемые пользователем.
3. Алгоритм проверки внешних устройств. Данный алгоритм позволяет отследить изменения в аппаратной части персонального компьютера, а также производить проверку устройств через интервалы времени выбираемые пользователем.
4. Вывод результатов проверки. Выводится сообщение об изменении или не изменении файлов и устройств персонального компьютера на основе чего оператор принимает решение о целостности данных системы.
5. Анализ результатов. В данном блоке пользователь принимает решение на основе выданного сообщения.

Поясним подробнее алгоритм проверки внешних устройств. Схема его приведена на рисунке 2.

Поясним вышеуказанную схему алгоритма. В ней можно отметить следующие основные операции.

1. Проверка подключенных устройств. Проводится анализ конфигурации компьютера по всем подключенным устройствам и программному обеспечению на данном компьютере.
2. Создание эталонного файла. Из конфигурации компьютера копируется информация о всех подключенных устройствах на данный момент времени в текстовый файл именуемый «Эталонным».
3. Создание файла сравнения. Для выяснения изменения структуры подключенных устройств по истечению некоторого времени повторяется последовательность действий описанных в пункте 2. За исключением того, что информация сохраняется в текстовый файл «Файл сравнения».
4. Сравнение двух файлов: Эталонного и Файла сравнения. В этом блок происходит сравнение двух текстовых файлов для определения, изменялись ли устройства за прошедший интервал времени.
5. Выдача сообщения о добавлении нового устройства (или нескольких устройств) или сообщения о том, что состояние системы не изменилось.

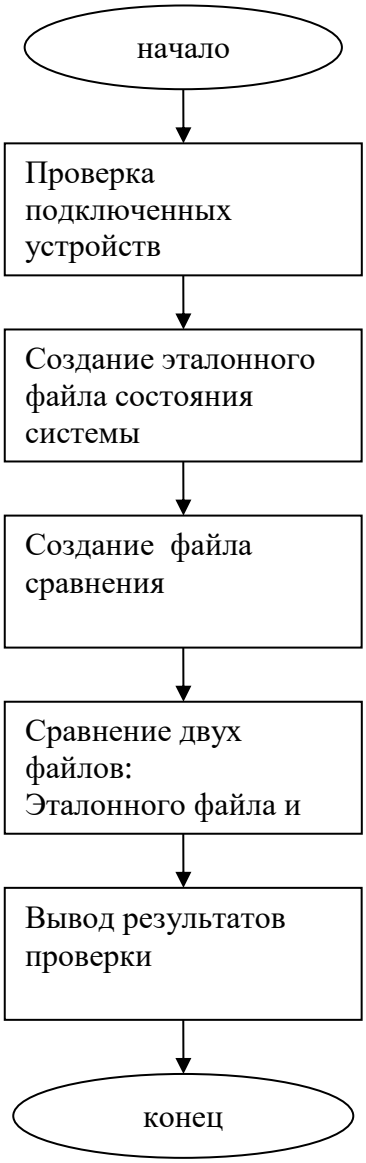


Рисунок 2. Алгоритм проверки подключенных устройств

## Описание программного продукта

На основе алгоритма расчета контрольной суммы и алгоритма проверки внешних устройств разработан комплексный алгоритм защиты целостности данных компьютера. С использованием рассмотренного алгоритма построена программа. Вся программа разделена на две части: устройства и контрольные суммы. Переход между окнами осуществляется при помощи аналогичных вкладок. На рисунке 3 изображена программа для расчета контрольной суммы.

На рисунке 4 изображено окно программы проверки внешних устройств.

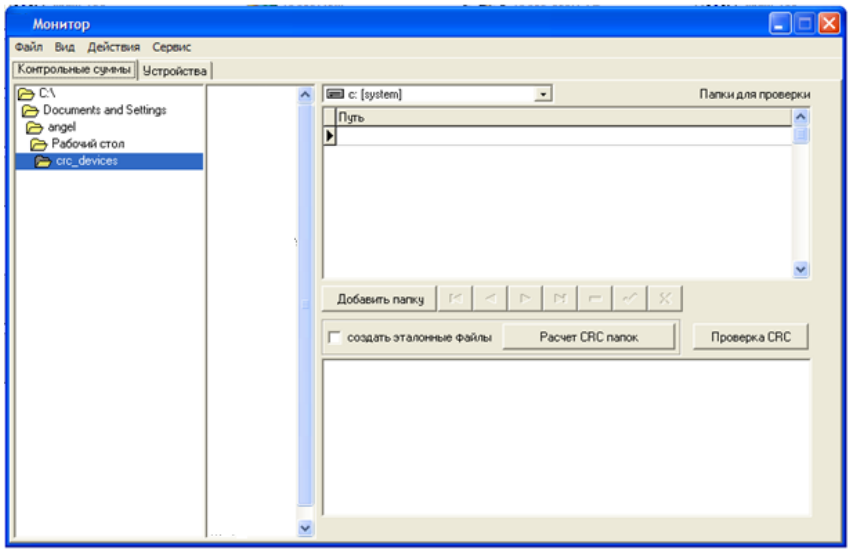


Рисунок 3. Окно расчета контрольной суммы

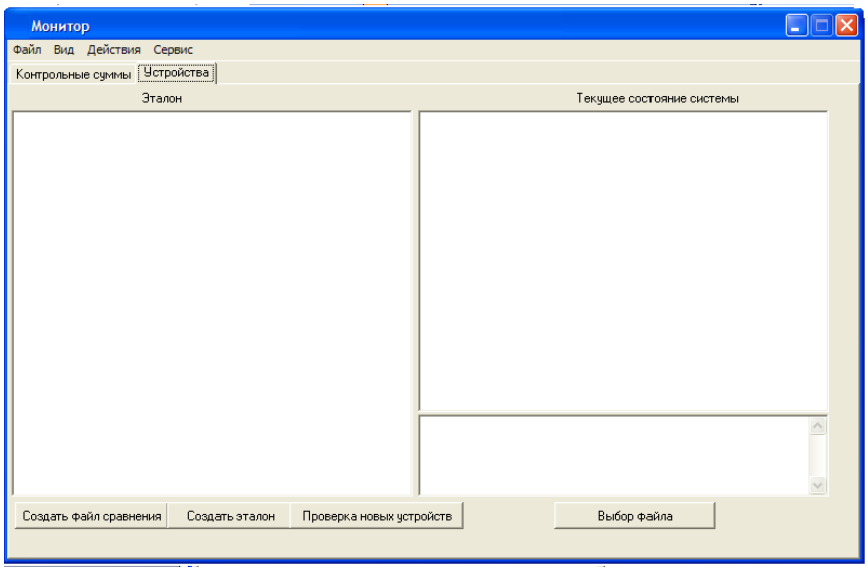


Рисунок 4. Окно проверки внешних устройств

В окне программы контрольные суммы можно проводить расчет контрольной суммы, как файла, так и всех файлов в папке. Расчет контрольной суммы файла происходит после двойного клика мыши на файл в указанной директории (рис. 5).

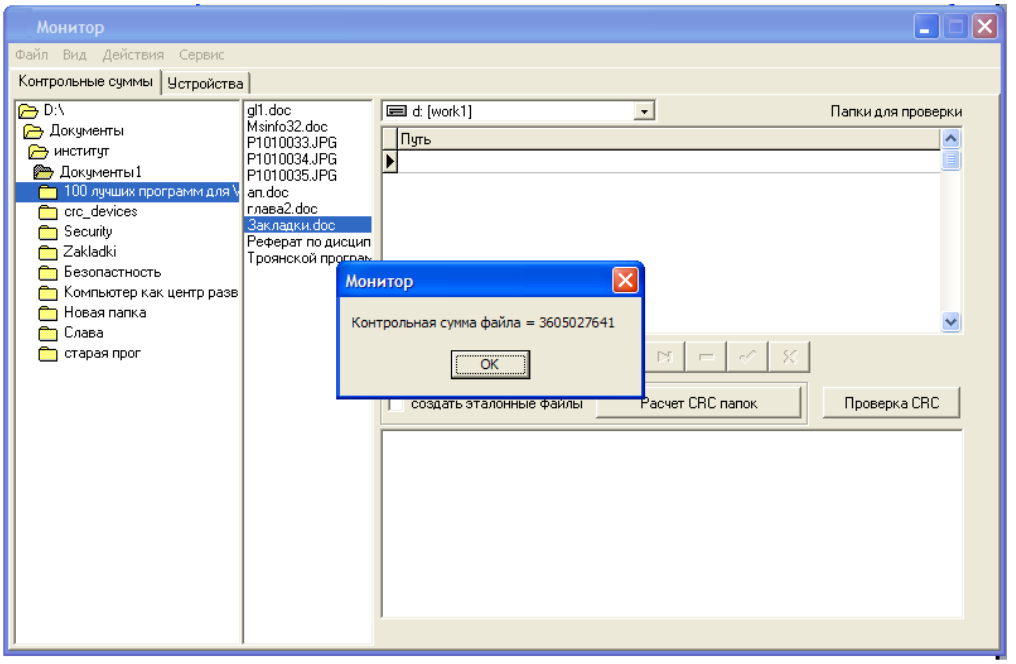


Рисунок 5. Окно расчета контрольной суммы файла

Так же в этом окне можно рассчитать контрольную сумму всех файлов в папке и занести папку в меню программы “Папки для проверки” для дальнейшей проверки целостности данных хранящихся в этой папке. Для проверки целостности данных хранящихся в папке необходимо:

1. Выбрать нужную папку в директории.
2. Нажать кнопку “Добавить папку”.
3. Поставить галочку “Создать эталонные файлы”.
4. Нажать кнопку “Расчет CRC папки”.

После чего будет произведен расчет контрольной суммы всех файлов в папке и это значение будет запомнено программой. Это позволит произвести проверку всех файлов в папке через некоторый интервал времени, выбираемый пользователем. Для проверки контрольной суммы всех файлов в папке необходимо нажать кнопку “Проверка CRC” и программа выдаст сообщение о совпадении или не совпадении контрольной суммы папки. Пример работы программы приведен на рисунке 6.

Удаление или изменение хотя бы одного файла приведет к изменению контрольной суммы и программа при следующей проверки выведет предупреждающее сообщение рисунке 7.

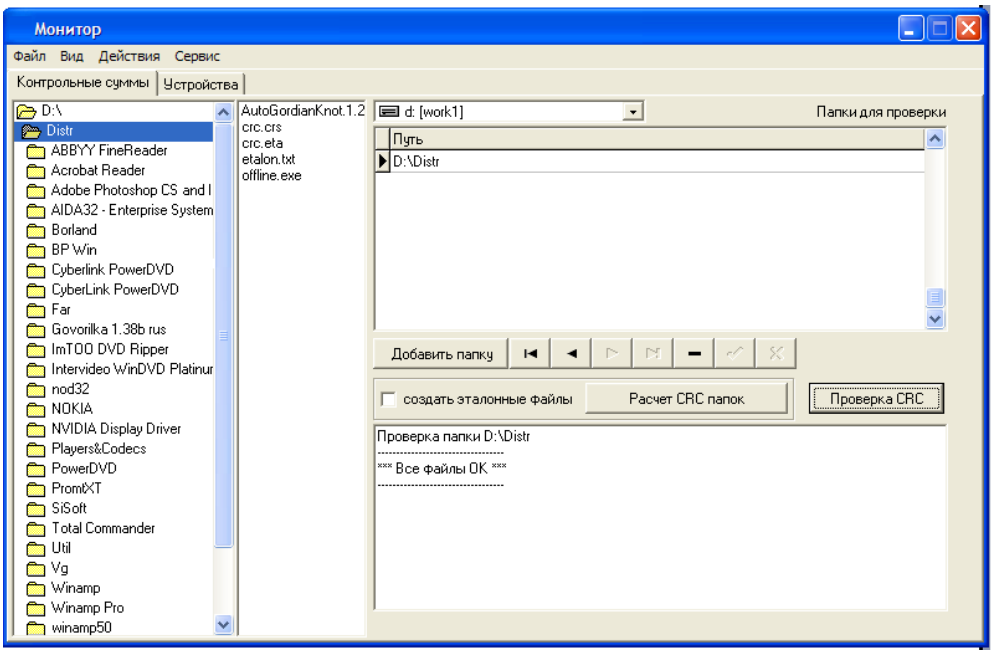


Рисунок 6. Пример расчета CRC всех файлов в папке

Для проверки внешних устройств необходимо перейти к вкладе “Устройства”. После чего создать эталонный файл с состоянием системы для этого необходимо произвести последовательность действий:

1. Нажать кнопку “Создать эталон”.
2. Проделать действия по подсказкам программы.
3. Аналогично создать файл сравнения.
4. Нажать кнопку “Выбор файла” и выбрать файл с именем ETALON.txt. Повторить нажатие кнопки “Выбор файла” и выбрать файл FILESRAVNENIA.txt
5. Нажать кнопку “Проверка новых устройств”.

После таких действий будет выдано сообщение (см. рис. 8).

Для проверки через любой интервал времени требуется пересоздать FILESRAVNENIA.txt и осуществить действия 4,5 описанные выше. Если в аппаратной части ПЭВМ произошли изменения, программа выдаст сообщение (рис. 9).

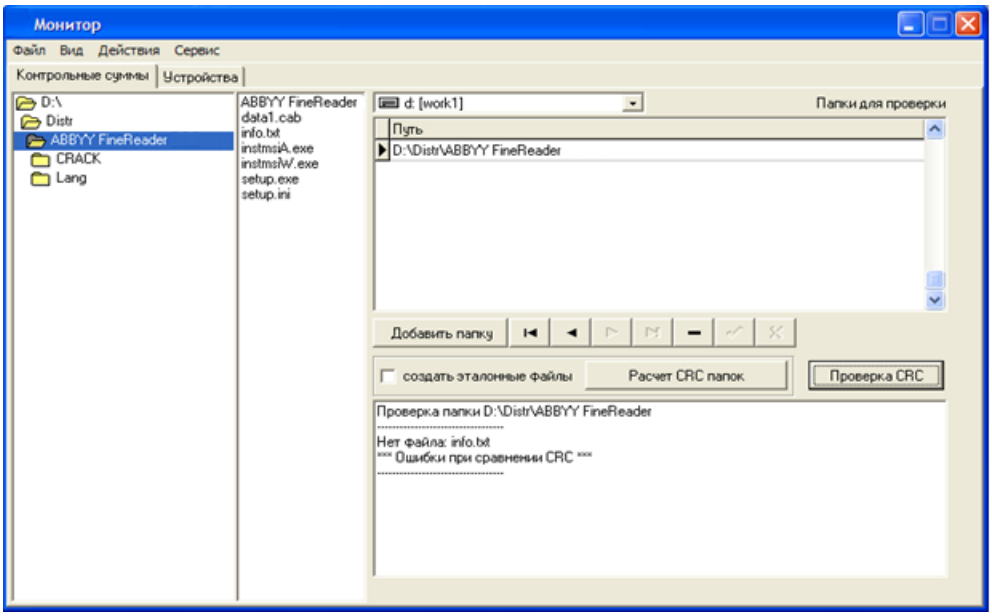


Рисунок 7. Пример работы программы при ошибке в файле

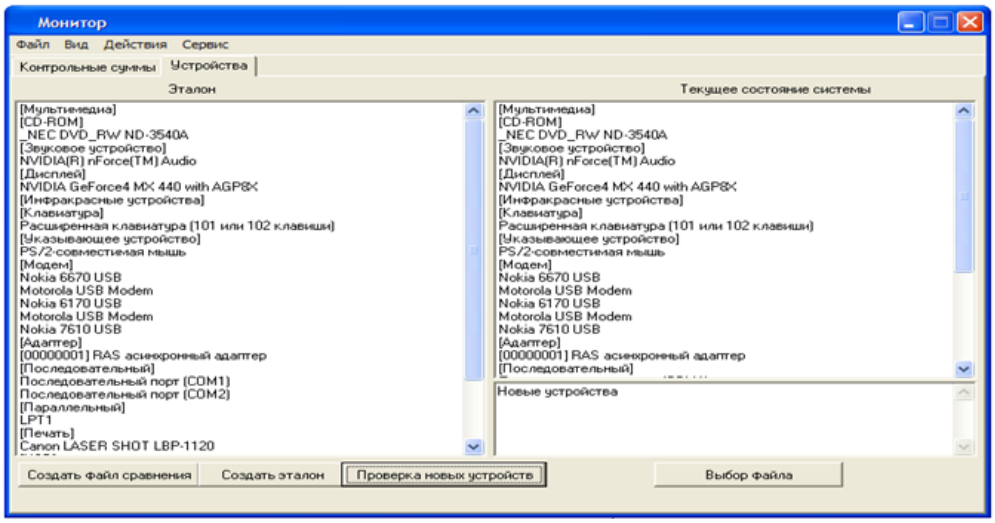


Рисунок 8. Пример работы программы

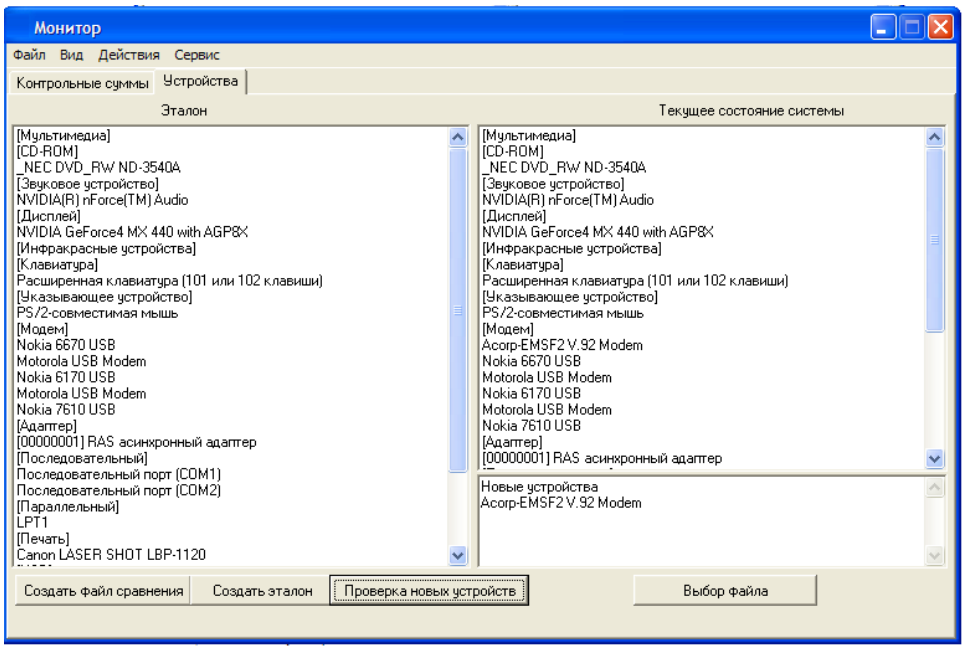


Рисунок 9. Пример работы программы при добавлении нового устройства

## Выводы

Работа посвящена разработке алгоритма и программного средства, позволяющего отслеживать изменения в программной и аппаратной части компьютера. Проведен анализ современных подходов к анализу целостности данных. Проанализированы достоинства и недостатки этих подходов. Даны рекомендации по построению алгоритма целостности данных на основе комбинации аппаратного и программного подходов. Дано описание программного продукта, созданного на основе алгоритма контрольной суммы и алгоритма проверки внешних устройств, позволяющего отслеживать изменения в программной и аппаратной части компьютера. Достоинством программного продукта является то, что в нем комплексным образом проводится анализ программной и аппаратной целостности данных компьютера.

## Литература

[1] Преображенский Ю. П. Некоторые аспекты информатизации образовательных учреждений и развития медиакомпетентности преподавателей и руководителей / Ю. П. Преображенский, Н. С. Преображенская, И. Я. Львович // Вестник Воронежского государственного технического университета. –2013. – Т. 9. – № 5-2. – С. 134-136.

[2] Воронов А. А. Обеспечение системы управления рисками при возникновении угроз информационной безопасности А. А. Воронов, И. Я. Львович, Ю. П. Преображенский, В. А. Воронов // Информация и безопасность. – 2006. – Т. 9. – № 2. – С. 8-11.

[3] Душкин А. В. Декомпозиционная модель угроз безопасности информационно-телекоммуникационным системам / А. В. Душкин, О. Н. Чопоров. // Информация и безопасность. – 2007. – Т. 10. – № 1. – С. 141-146.

[4] Сергеев А. В. Проблемы обнаружения и исправления ошибок в линиях связи / А. В.Сергеев, Х. И. Бешер, В. В. Кузнецов // Вестник Воронежского института высоких технологий. – 2016. – № 4 (19). – С. 22-24.

[5] Лобзин П. В. Особенности анализа целостности данных / П. В. Лобзин // Вестник Воронежского института высоких технологий. – 2016. – № 4 (19). – С. 96-98.

[6] Ермолова В. В. Методика построения семантической объектной модели / В. В. Ермолова, Ю. П. Преображенский // Вестник Воронежского института высоких технологий. – 2012. – № 9. – С. 87-90.

[7] Преображенский Ю. П. Алгоритм нахождения оптимальной стационарной стратегии для марковских процессов принятия решений / Ю. П. Преображенский // Вестник Воронежского института высоких технологий. – 2010. – № 6. – С. 81-82.

[8] Фомина Ю. А. Принципы индексации информации в поисковых системах / Ю. А. Фомина, Ю. П. Преображенский // Вестник Воронежского института высоких технологий. – 2010. – № 7. – С. 98-100.

[9] Ермилов Е. В. Риск-анализ распределенных систем на основе параметров рисков их компонентов / Е. В. Ермилов, Е. А. Попов, М. М. Жуков, О. Н. Чопоров // Информация и безопасность. – 2013. – Т. 16. – № 1. – С. 123-126.

[10] Глотова Т. В. Анализ подходов, обеспечивающих защищенность современных компьютерных систем / Т. В. Глотова, Х. И. Бешер // Моделирование, оптимизация и информационные технологии. – 2016. – № 3 (14). – С. 17.

[11] Преображенский Ю. П. Оценка эффективности применения системы интеллектуальной поддержки принятия решений / Ю. П. Преображенский // Вестник Воронежского института высоких технологий. – 2009. – № 5. – С. 116-119.

[12] Зяблов Е. Л. , Преображенский Ю. П. Построение объектно-семантической модели системы управления Е. Л. Зяблов, Ю. П. Преображенский // Вестник Воронежского института высоких технологий. – 2008. – № 3. – С. 029-030.

[13] Преображенский Ю. П. Разработка методов формализации задач на основе семантической модели предметной области / Ю. П. Преображенский // Вестник Воронежского института высоких технологий. – 2008. – № 3. – С. 075-077.

[14] Пахомова А. С. Целенаправленные угрозы компьютерного шпионажа: признаки, принципы и технологии реализации / А. С. Пахомова, О. Н. Чопоров, К. А. Разинкин // Информация и безопасность. – 2013. – Т. 16. – № 2. – С. 211-214.

[15] Пахомова А. С. Атаки на информационно-технологическую инфраструктуру критически важных объектов: оценка и регулирование рисков: монография / А. С. Пахомова, О. Н. Чопоров, К. А. Разинкин; под ред. чл.-корр. РАН Д. А. Новикова. – Воронеж, Издательство: ООО «Издательство «Научная книга», 2013. – 159 с.

[16] Мэн Ц. Анализ методов классификации информации в интернете при решении задач информационного поиска / Ц. Мэн // Моделирование, оптимизация и информационные технологии. – 2016. – № 2 (13). – С. 19.

……

**Prof. Oleg Choporov**
Doctor of Sciences (Engineering), Professor,
Voronezh Institute of High Technologies
E-mail: choporov_oleg@mail.ru

**Prof. Igor Lvovich**
Doctor of Sciences (Engineering), Professor,
Voronezh Institute of High Technologies
E-mail: i_lvovich@mail.ru

**Konstantin Razinkin**
Doctor of Sciences (Engineering), Associate Professor,
Voronezh Institute of High Technologies
E-mail: kostyr@mail.ru