**Министерство науки и высшего образования Российской Федерации  
ФГБОУ ВО «МГУТУ им. К. Г. Разумовского (ПКУ)»  
Университетский колледж информационных технологий**

Отчет по учебной практике   
УП.04.01. Учебная практика по профессиональному модулю

ПМ.04 «Сопровождение и обслуживание программного обеспечения компьютерных систем»,   
студента группы 090207-9ои-21/0 Шахова Андрея Дмитриевича,

обучающегося специальности   
09.02.07 «Информационные системы и программирование»

Выполнил студент: Шахов А. Д.  
Преподаватель: Плахутина Л. А.

Москва  
2023

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc17344)

[ОСНОВНАЯ ЧАСТЬ 5](#_Toc9002)

[ЗАКЛЮЧЕНИЕ 20](#_Toc17989)

[ПРИЛОЖЕНИЕ 1 21](#_Toc4891)

# ВВЕДЕНИЕ

Выполнение заданий Учебной практики 04.01 ПМ.04. Сопровождение и обслуживание программного обеспечения компьютерных систем подразумевает освоение ряда умений и компетенций.   
 Данная учебная практика формирует следующие профессиональные компетенции:   
 ПК 4.1. Осуществлять инсталляцию, настройку и обслуживание программного обеспечения компьютерных систем.   
 ПК 4.2. Осуществлять измерения эксплуатационных характеристик программного обеспечения компьютерных систем.   
 ПК 4.3. Выполнять работы по модификации отдельных компонентов программного обеспечения в соответствии с потребностями заказчика.   
 ПК 4.4. Обеспечивать защиту программного обеспечения компьютерных систем программными средствами.

В ходе работы надо получить практический опыт:   
 - в настройке отдельных компонентов программного обеспечения компьютерных систем;   
 - выполнении отдельных видов работ на этапе поддержки программного обеспечения компьютерной системы.   
Во время учебной практики должны быть приобретены навыки:   
 - подбирать и настраивать конфигурацию программного обеспечения компьютерных систем;

- использовать методы защиты программного обеспечения компьютерных систем;   
 - проводить инсталляцию программного обеспечения компьютерных систем;   
 - производить настройку отдельных компонентов программного обеспечения компьютерных систем;   
 - анализировать риски и характеристики качества программного обеспечения.   
Помимо этого, были получены общие компетенции:   
 ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.   
 ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.   
 ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие.   
 ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.   
 ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.   
 ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей.

ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.

ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.

ОК 09. Использовать информационные технологии в профессиональной деятельности.

ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке.

ОК 11. Планировать предпринимательскую деятельность в профессиональной сфере.

# ОСНОВНАЯ ЧАСТЬ

Данная учебная практика содержит 52 практические работы, которые выполняются на компьютере.

Практические работы №1, 10, 11 позволили получить навыки работы с документами для внедрения программного продукта. Задача заключалась в разработке сценария внедрения программного продукта, разработке отчетных форм для внедрения программных средств, разработке руководства оператора для внедрения программных средств.

Внедрение программного обеспечения - процесс настройки программного обеспечения под определенные условия использования, а также обучения пользователей работе с программным продуктом. Составление сценария состоит из 5 этапов: обследование компании, составление контракта на производство работ, создание группы по внедрению программного обеспечения (ПО), инсталляция и наладка ПО, завершение внедрения.

Оператор по своим функциям ближе всего к роли пользователя, однако, отличается от нее тем, что перед оператором не ставятся прикладные задачи, которые он может решить с помощью программы. Его работа заключается в выполнении отдельных операций, то есть конкретных последовательностей действий, приводящих к конкретному результату (например, ежедневный запуск вспомогательных программ).

Документацию для отчетных форм можно разбить на две группы:

1. Документы управления разработкой программного средства (ПС);
2. Документы, входящие в состав программного средства (ПС).

Документы управления разработкой ПС управляют процессами разработки и сопровождения ПС, обеспечивая связи внутри коллектива разработчиков ПС и между коллективом разработчиков и менеджерами. Эти документы могут быть следующих типов: планы, оценки, расписания, отчеты об использовании ресурсов в процессе разработки, стандарты, рабочие документы.

В практических работах №2 - 3 задача заключалась в изучении функций менеджера сопровождения и менеджера развертывания.

Менеджер сопровождения включает в себя:

1. Управление информационной системой (ИС), в том числе параметрами настройки ПО и при необходимости восстановление их работоспособности;
2. Установка обновлений программного обеспечения, в том числе программное обеспечение ИС;
3. Изменение эксплуатационной документации и организационно-распорядительных документов по защите информации;
4. Корректировки базовой конфигурации ИС и системы ее защиты;
5. Проведение повторной аттестации ИС или дополнительных аттестационных испытаний при необходимости;
6. Периодический контроль уровня защищенности информации в ИС;
7. Периодический контроль эффективности работы ИС, в том числе устранение ошибок и недостатков в ее функционировании;
8. Периодический контроль изменения угроз безопасности информации в процессе эксплуатации ИС и оперативное принятие мер защиты информации при возникновении новых угроз.

Сопровождение информационной системы позволяет максимально оперативно решать вопросы, которые могут возникнуть во время её использования, при изменении самой системы (частично или полностью).

Менеджер (диспетчер) развертывания - это компонент, который управляет запуском и завершением работы блоков обработки в зависимости от условий нагрузки. Он запускает новые процессорные блоки, когда пользовательская нагрузка увеличивается, и выключает процессорные блоки, когда пользовательская нагрузка уменьшается.

В практических работах №4 - 7 необходимо автоматизировать процесс внедрения информационной системы с помощью типовых функций инструментария, описать оценку качества функционирования информационной системы и оценку качества функционирования информационной системы с помощью CALS-технологий, а также провести обновление информационной системы.

Качество информационной системы связано с дефектами, заложенными на этапе проектирования и проявляющимися в процессе эксплуатации. Свойства информационной системы, в том числе и дефектологические, могут проявляться лишь во взаимодействии с внешней средой, включающей технические средства, персонал, информационное и программное окружение.

В практических работах №8, 9, 30, 31, 35, 48, 50 провели тестирование программного обеспечения в процессе внедрения, эксплуатации, тестирование программных продуктов, защиты программного обеспечения, тестирование программного продукта по принципам “10 причин Майерса”, “черного” и “белого” ящика, изучили первичные и вторичные ошибки и провели анализ результатов тестирования.

Тестирование проводится снизу вверх, то есть сначала тестируются отдельные функции, затем целые модули и далее проводится комплексное тестирование всей программы или комплекса программ. Для проведения тестирования разрабатывается тест-план - совокупность тестовых наборов. В каждом тестовом примере производится выполнение тестируемого программного элемента (SUT) при заданных Input - условиях и входных данных и проверяются все Output - выходные данные на соответствия заданным значениям. Тестовый пример должен включать в себя как минимум:

1. Входы (конкретные значения всех выходных параметров, все необходимые свойства и установки окружения);
2. Действия (что надо выполнить и в какой последовательности);
3. Ожидаемый выход (конкретные величины всех возвращаемых значений).  
    Для проведения тестирования разрабатывается программа-драйвер, выполняющая все тестовые примеры и сравнивающая выходные значения с ожидаемыми. В результате выполнения теста получается не только общий результат, но еще и список пройденных и непройденных тестовых примеров, который помогает локализовать ошибки в SUT.  
    Для упрощения локализации ошибок и последующей модификации тест-плана нужно, чтобы тестовые примеры были независимы друг от друга, то есть чтобы каждый последующий тестовый пример никак не использовал результаты работы предыдущего. Для этого необходимо провести установки всех начальных условий перед выполнением каждого тестового примера.

Тестирование белого ящика подразумевает, что у разработчика теста есть доступ к исходному коду приложения и он имеет возможность писать код, связанный с библиотеками тестируемого ПО. Такое положение дел часто встречается при юнит-тестировании. В этом случае проверке подвергаются лишь определенные элементы системы. Ниже приведены 10 причин Майерса для тестирования:

1. Обязательная часть тестирования - определение ожидаемого результата;
2. Программистам следует избегать тестирования их собственных программ (и участков кода);
3. Организациям, создающие программы, следует избегать тестирования их собственных программ;
4. Процесс тестирования должен включать в себя тщательную проверку результатов каждого теста;
5. Тест-кейсы должны быть составлены как для корректных и ожидаемых входных условий, так и для некорректных и неожидаемых;
6. Исследование системы на предмет того, что она не делает, что должна - лишь наполовину;
7. Одноразовые тест-кейсы;
8. Тестирование с предустановкой приводит к ошибке;
9. Вероятность наличия ошибок в определенной части системы пропорционально количеству уже найденных ошибок;
10. Тестирование - это вызов творческим и интеллектуальным способностям.

При сборе требований необходимо учесть мнение клиентов. Это нужно для того, чтобы определить реальные и предполагаемые результаты тестирования, которые чаще всего являются нефункциональными.

На данной стадии тестировщики рассматривают и анализируют требования, а также формируют соответствующие тесты. Кроме того, они определяют приоритеты для проверки - членов команды.

В список требований к среде тестирования входят требования к аппаратному и программному обеспечению. На их основе нужно будет выполнять проверку ПО. Одновременно с этим начинаются планирование и разработка программного обеспечения.

Тестирование безопасности - это тип тестирования ПО, который выявляет уязвимости, угрозы и риски. Целью тестов безопасности является выявление всех возможных лазеек и слабых мест в ПО, которые могут привести к потере информации, доходов, репутации компании, сотрудников или клиентов.

Тестирование программного обеспечения - проверка соответствия реальных и ожидаемых результатов поведения программы, проводимая на конечном наборе тестов, выбранном определенным образом. Более того, в процессе отладки и тестирования, в первую очередь выявляются вторичные ошибки, связанные с проявлением исходных дефектов, которые следует квалифицировать как первичные ошибки. В свою очередь, локализация и корректировка первичных ошибок приводит к устранению первоначально обнаруживаемых ошибок.

Практические работы №12, 22, 23 основаны на восстановлении системы, диагностики оборудования, оптимизации использования процессора, памяти, жесткого диска, сети. Решение проблем аппаратного сбоя. Изучение чистой загрузки.

"Чистая загрузка" запускает Windows с минимальным набором драйверов и программ запуска, чтобы можно было определить, влияет ли фоновая программа на вашу игру или программу.

Для перехода в чистую загрузку необходимо перейти на вкладку "Службы" в разделе "Конфигурация системы" выбрать "Скрыть все службы Майкрософт", а затем выбрать "Отключить все", нажав кнопку "Применить" (Приложение 1, рисунок 1). Далее на вкладке "Запуск" в разделе "Конфигурация системы" выбрать "Открыть диспетчер задач". Далее на вкладке "Запуск" в диспетчере задач для каждого включенного элемента запуска выбраны приложения которые нужно отключить. После перезапуска компьютер загрузился в режиме чистой загрузки.

Восстановление системы делает "моментальный снимок" некоторых системных файлов и реестра Windows и сохраняет их в качестве точек восстановления. При сбое установки или повреждения данных восстановление системы может вернуть систему в рабочее состояние без необходимости переустановки операционной системы (Приложение 1, рисунок 2).

Для оптимизации процессора можно распределить загрузку ядер. Все ядра процессора работают с разной частотой (одновременно), и задействуются в полную мощь тогда, когда это требуется. В [повседневных задачах](https://mkr-novo2.ru/education/planshet-explay-hit-obzor-harakteristik-i-osnovnyh-funkcii-planshet.html) они работают в [обычном режиме](https://mkr-novo2.ru/navigation/kak-vernutsya-v-obychnyi-rezhim-na-androide-kak-otklyuchit-bezopasnyi-rezhim.html). Это даёт возможность достижения баланса производительности, а значит, устройство или его компоненты не выйдут из строя раньше времени. Для удобной оптимизации ядер использовано приложение CPU-Z (Приложение 1, рисунок 3) с помощью которого распределена нагрузка на отдельные ядра процессора, что привело к повышению производительности компьютера.

Для оптимизации использования памяти компьютера можно: очистить кэш ОЗУ, обновить драйвера, установить системные обновления, проверить систему на наличие вирусов, отключить программы автозагрузки и фоновых приложений.

Оптимизировать сеть можно подключив новое сетевое оборудование без изменений в архитектуре, установив дополнительные модули и лицензии на существующее оборудование либо провести дополнительную настройку оборудования.

Практические работы №13, 14, 17, 18 позволили приобрести навыки в определении совместимости программного обеспечения отраслевой направленности с и операционными системами. Позволили ознакомиться с мастером совместимости программ.

Совместимость - способность аппаратных и программных средств работать с аппаратным обеспечением компьютерной системы, а также способность компонентов взаимодействовать друг с другом.

Причины возникновения проблем совместимости - несовершенство программного обеспечения, несовершенство операционной системы, отсутствие нужных ресурсов, ошибки в реестре, конфликты между устройствами, вирусы.

Аппаратное обеспечение не бывает полностью совместимо и может отказаться работать. Например устранить проблему неисправности звуковой карты или привода компакт-диска можно с помощью диспетчера устройств. Либо проанализировать инструкцию к устройству. Совместимость восстанавливается если: переустановить плату расширения в другой слот, обновить ПО, заменить устройство более новой моделью.

Методы выявления проблем совместимости ПО заключаются в обязательной проверке используемого ПО на совместимость с новой ОС и сбор информации о выявленных в процессе этого тестирования проблемах.

Практические работы №15, 19 заключались в выявлении и документировании причин возникновения проблем совместимости ПО.

Информационный процесс является разновидностью технологического процесса, содержащего действия по изменению состояния информации.

На основе информационных технологий решается задача автоматизации информационных процессов.

После установки группы необходимых исправлений совместимости, нужно воспользоваться администратором совместимости, для устранения неполадок.

Администратор совместимости, поставляемый с операционной системой Windows, может быть найден в папке «Support Tools» на установочном компакт-диске.

Практическая работа № 16 заключалась в обновлении версии ПО компьютера.

Обновления - это дополнения к программному обеспечению, предотвращающие или устраняющие неполадки, повышающие безопасность либо улучшающие производительность компьютера. Обновление приходит от разработчика в любое время и несет разные изменения. Установка происходит автоматически или предлагается установить сейчас. например ПО Windows предлагает на выбор установить обновление потом или сейчас (Приложение 1, рисунок 4).

Практическая работа №20 позволила приобрести опыт работы с VirtualBox и заключалась в создании образа системы в виртуальной машине.

Для переноса физических компьютеров с операционными системами Windows, на виртуальный жесткий диск формата VHD или VHDX, с целью дальнейшего использования в Microsoft Virtual PC или Hyper-V была использована утилита Disk2vhd. С помощью данной утилиты можно "виртуализировать" компьютер с Windows. Для этого:

1. Скачена программа Disk2vhd;
2. В программе Disk2vhd, в поле "VHD File name", выбрано место сохранения для виртуального жесткого диска. Ниже выбраны необходимые для сохранения разделы жесткого диска (Приложение 1, рисунок 5);
3. По завершению создания виртуального жесткого диска, он был скопирован на компьютер с VirtualBox;
4. Для создания виртуальной машины в VirtualBox использован существующий виртуальный диск. (Приложение 1, рисунок 6);
5. Запуск виртуальной машины и загрузка в рабочую систему.

Практические работы №21, 17 заключались в тестировании на совместимость в безопасном режиме, изучении объекта уязвимости программного продукта, дестабилизирующих факторов и угрозы надежности программного продукта и методов предотвращения угроз надежности программного продукта.

Внутренние источники угроз надежности функционирования сложных программных систем (ПС) присущи самим объектам уязвимости: ошибки проектирования при постановке задачи, алгоритмические ошибки разработки при спецификации функций ПС, при определении структуры и взаимодействия компонент комплексов программы, а также при использовании информации баз данных.

Внешние дестабилизирующие факторы обусловлены средой, в которой функционируют объекты уязвимости. Ими являются ошибки персонала при эксплуатации, искажение информации в каналах связи, сбои и отказы аппаратуры, изменение конфигурации аппаратуры информационной системы.

Основные методы предотвращения угроз надежности: предотвращение ошибок проектирования в CASE - технологиях, систематическое тестирование, обязательная сертификация. Комплексное применение этих методов позволяет значительно ослаблять влияние угроз. Таким образом, уровень достигнутой надежности зависит от ресурсов, выделяемых на его достижение и от качества технологий, используемых на всех этапах жизненного цикла программного средства.

В безопасном режиме запуск Windows выполняется с использованием ограниченного набора файлов и драйверов. В безопасном режиме не производится запуск программ, загружаемых при запуске, и устанавливается только базовый набор драйверов, необходимый для запуска Windows.

Безопасный режим полезен для устранения неполадок, вызванных программами и драйверами, которые могут неправильно загружаться или препятствовать правильному запуску Windows. Если при загрузке в безопасном режиме эти проблемы не возникают, то из списка возможных причин можно исключить параметры, используемые по умолчанию, и базовый набор драйверов устройств. Объектами уязвимости являются:

1. Динамически вычислительный процесс обработки данных, автоматизированной подготовки решений и выработки управляющих воздействий на потребителей обработанной информации;
2. Информация, накопленная в базах данных, отражающая объекты внешней среды, и процессы ее обработки;
3. Объектный код программ, исполняемых вычислительными средствами в процессе функционирования ПС;
4. Информация, выдаваемая потребителям и на исполнительные механизмы, являющаяся результатом обработки исходных данных и информации, накопленной в базе данных;

Практическая работа №24 позволила приобрести навыки установки сервера и работы с ним. Задание заключалось в установке сервера.

Серверная часть - это виртуальный центр связи, который соединяется с компьютерами и передаёт данные с одного компьютера на другой. Серверная часть ставится на сервере - для этого следует использовать учебный сервер для виртуального хранилища.

Выполнение работы происходило последовательно. Сначала происходила установка сервера. Далее проходила настройка сервера (Приложения 1, рисунок 7, 8).

Практическая работа №26 заключалась в изучении модели качества программного продукта.

Модель качества продукта сводит свойства качества системы/программного продукта к восьми характеристикам, которыми являются: функциональная пригодность, уровень производительности, совместимость, удобство пользования, надежность, защищенность, сопровождаемость и переносимость (мобильность). Каждая характеристика, в свою очередь, состоит из ряда соответствующих подхарактеристик (Приложение 1, рисунок 9).

Модели качества продукции и качества при использовании могут быть использованы для определения требований, выработки показателей и выполнения оценки качества. Определенные характеристики качества могут использоваться в качестве контрольного списка для обеспечения детального исследования требований к качеству, обеспечивая таким образом основу для оценки необходимых в процессе разработки систем последующих трудозатрат и действий. Характеристики в модели качества при использовании и модели качества продукта предназначены для использования в качестве набора при спецификации или оценке качества программного продукта или компьютерной системы.

Практические работы №25, 38 позволили приобрести навыки установки клиентского программного обеспечения с его адаптацией. Задание заключалось в установке клиентского ПО (Приложение 1, рисунок 10) с внедрением в него модуля адаптации (Приложение 1, рисунок 11).

Клиентская часть устанавливается и запускается на компьютерах, где нужен удаленный доступ для управления.

Сначала был разработал модуль (Приложение 1, рисунок 12). Далее было разработано ПО и модуль адаптации (Приложение 1, рисунок 10). После этого модуль адаптации был внедрен в мое программное обеспечение (Приложение 1, рисунок 13).

Практические работы №32 - 34 заключали в себя изучение методов повышения надежности временной избыточности, информационной избыточности и программной избыточности.

Временная избыточность - заключается в использовании некоторой части производительности компьютера для контроля за исполнением программ и восстановления вычислительного процесса например, двойного или тройного просчёта на вычислительной машине.

Информационная избыточность состоит в дублировании исходных и промежуточных данных обрабатываемых программами. Избыточность используется для сохранения достоверности данных, которые в наибольшей степени влияют на нормальное функционирование ПС и требуют значительного времени на восстановление.

Программная избыточность используется для контроля и обеспечения достоверности наиболее важных решений ПО обработке информации. Она заключается в сопоставлении результатов обработки одинаковых исходных данных. Программная избыточность необходима также для реализации программ автоматического контроля и восстановления данных с использованием информационной избыточности и для функционирования всех средств обеспечения надежности, использующих временную избыточность.

Практические работы №36, 37, 40 позволили приобрести навыки настройки обновлений с помощью зеркала его адаптацию и его сопровождения. А также провести анализ рисков и характеристик программного продукта. Задание заключалось в настройке обновлений с помощью зеркала с анализом программного продукта (Приложение 1, рисунок 14, 15).

Зеркало обновлений - это папка, в которую копируются обновления.

Риски характеризуют возможные негативные последствия или ущерб при функционировании ПС и системы, и задача разработчиков сводится к сокращению и ликвидации рисков.

При обеспечении качества программных систем одной из наиболее приоритетных задач является формализация, стандартизация показателей качества и эффективные методологии их оценки. Это означает, что при подготовке технических заданий на разработку программных систем необходимо четко прописываются основные требования к характеристикам их качества с указанием, методов их измерения или экспертной оценки для того чтобы обеспечить наибольшую достоверность результатов тестирования и надлежащее сравнение с требованиями, заданными в контракте и спецификациях. Это позволит сократить вероятность конфликтов между заказчиками и пользователями с одной стороны и разработчиками и поставщиками с другой вследствие разной интерпретации одних и тех же показателей качества.

Практические работы №39, 41 позволили приобрести навыки обнаружение вирусов и решение проблем связанные с ними, а также установка антивируса. Задание заключалось в установке антивируса (Приложение 1, рисунок 16) с настройкой его параметров и вируса (Приложение 1, рисунок 17).

Компьютерный вирус - это программа, способная создавать свои копии, внедрять их в различные объекты систем и сетей и производить определенные действия без ведома пользователя.

Антивирусная программа это специализированная [программа](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) для обнаружения компьютерных [вирусов](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D1%8B%D0%B9_%D0%B2%D0%B8%D1%80%D1%83%D1%81), а также нежелательных программ и восстановления заражённых (модифицированных) такими программами [файлов](https://ru.wikipedia.org/wiki/%D0%A4%D0%B0%D0%B9%D0%BB) и профилактики предотвращения заражения (модификации) файлов или [операционной системы](https://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) вредоносным кодом.

Выполнение работы происходило последовательно. Изначально был установлен вирус, после был установлен бесплатный антивирус Avast. При этом была проведена проверка антивируса на борьбу с установленным мной вирусом.

Практические работы №42, 43 заключают в себя настройку браузера и политики безопасности.

Поскольку на компьютере, на котором проходило выполнение практических заданий стоит ОС Windows Home был создан командный файл gpedit.bat для установки необходимых компонентов работы с политикой безопасности. После скачивания дополнительных компонентов был произведен запуск редактора политики безопасности (Приложение 1, рисунок 18).

Для настройки безопасности браузера использовалось приложение «Безопасность Windows» в котором была открыта вкладка «Управление приложениями/браузером» (Приложение 1, рисунок 19) с помощью которого можно настроить защиту от ненадежных сайтов и приложений.

Практические работы №44 - 47 заключались в работе с реестром, учетными записями, изучении групповых политик и методы аутентификации.

Реестр содержит сведения, на которые Windows постоянно ссылается во время операции, такие как профили для каждого пользователя, приложения, установленные на компьютере, типы документов, которые могут создаваться, параметры таблицы свойств для папок и значков приложений, оборудование, которое установлено в системе, и используемые порты (Приложение 1, рисунок 20).

Групповая политика - важный элемент любой среды Microsoft Active Directory (AD). Ее основная цель - дать ИТ-администраторам возможность централизованно управлять пользователями и компьютерами в домене. Групповая политика, в свою очередь, состоит из набора политик, называемых объектами групповой политики (GPO).

Учетная запись пользователя (Приложение 1, рисунок 21) - это собрание информации, которое Windows использует для определения, к каким файлам и папкам имеет доступ пользователь и какие изменения на компьютере он может осуществлять. Также в нем хранятся личные настройки, например фон рабочего стола или заставка.

Практические работы №51, 52 позволили приобрести навыки шифрования данных и их методы расшифровки. Задание заключалось в зашифровке данных (Приложение 1, рисунок 22) и их расшифровки (Приложение 1, рисунок 23).

Тип сетевого протокола сеансового уровня (SOCKS) применяется в тех приложениях, которые либо не используют HTTP и HTTPS, либо не имеют встроенной поддержки прокси-серверов. В отличие от предыдущего типа, этот априори не будет публиковать IP.

VPN также изначально не задумывалась как средство защиты и анонимизации трафика. Ее задачей было объединить компьютеры в единую сеть, даже если они находятся за множество километров друг от друга. Ключевой особенностью стало то, что соединения VPN всегда защищались шифрованием, так как использовались в корпорациях и позволяли подключать несколько филиалов к головному офису.

Изначально был создан файл с текстом, после чего он был зашифрован с помощью программы на Visual Studio. Далее проведена расшифровка файла.

# ЗАКЛЮЧЕНИЕ

В результате прохождения учебной практики был получен практический опыт в настройке отдельных компонентов программного обеспечения компьютерных систем, выполнении отдельных видов работ на этапе поддержки программного обеспечения компьютерной системы, а также были приобретены навыки подбирать и настраивать конфигурацию программного обеспечения компьютерных систем, использовать методы защиты программного обеспечения компьютерных систем, проводить инсталляцию программного обеспечения компьютерных систем, производить настройку отдельных компонентов программного обеспечения компьютерных систем, анализировать риски и характеристики качества программного обеспечения.  
 Помимо этого, был получен опыт в работе с коллективом, способы выполнения профессиональных задач, оценены их эффективность и качество. Приобретен навык осуществление поиска и использования информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития, использование информационно-коммуникационных технологий в профессиональной деятельности.  
 Учебная практика позволила продвинуться в сфере сопровождения и обслуживания программного обеспечения компьютерных систем.

# **ПРИЛОЖЕНИЕ 1**

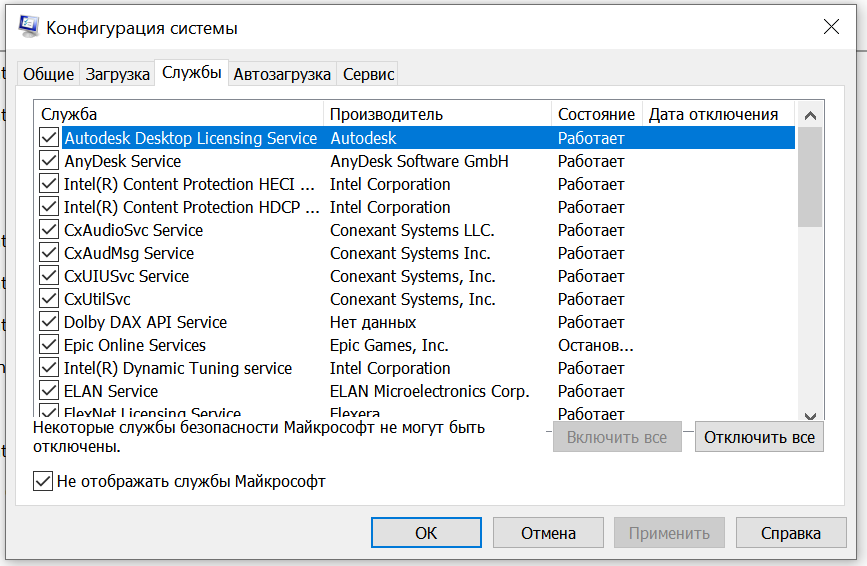


Рисунок 1 - Конфигурация системы

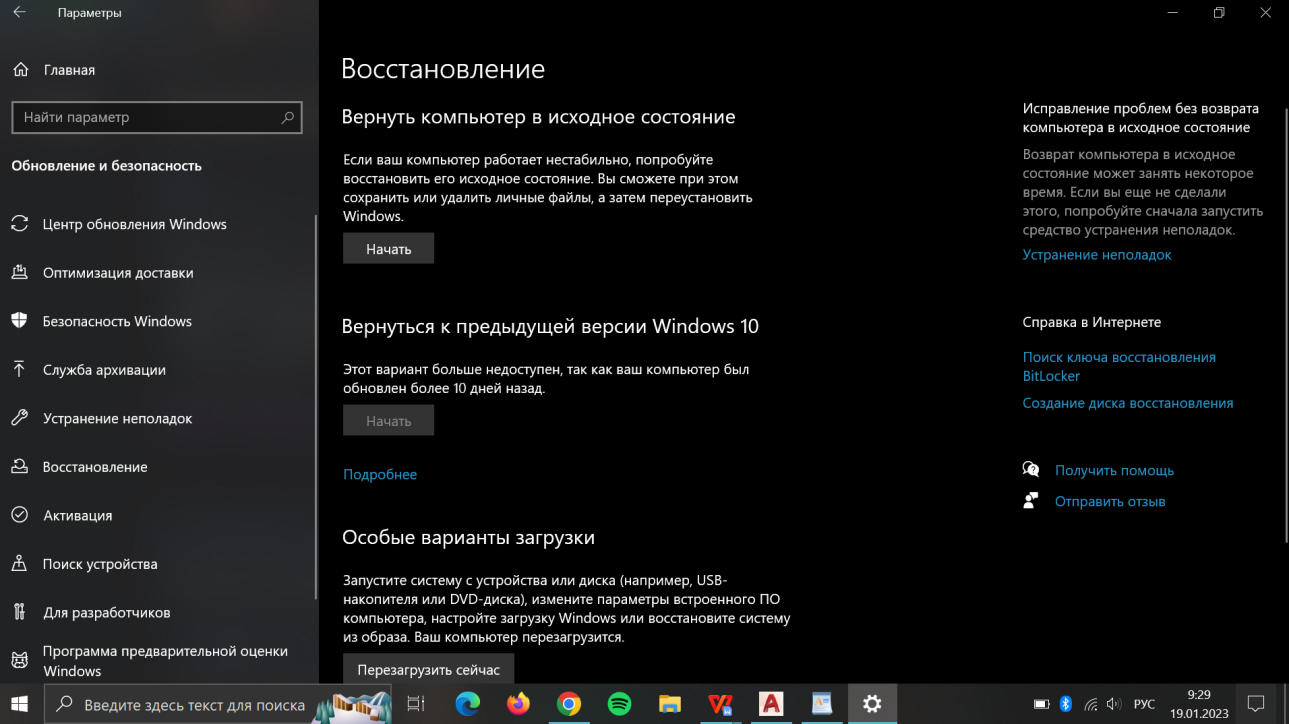


Рисунок 2 - Восстановление системы

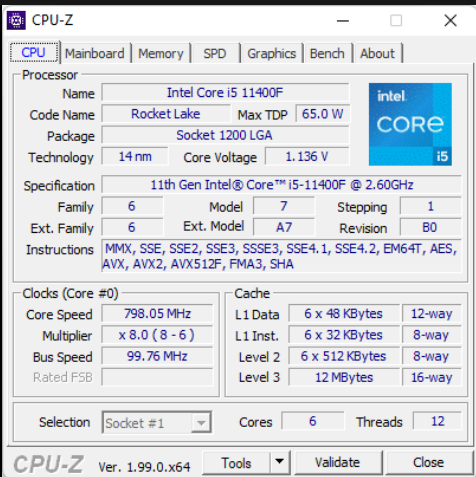


Рисунок 3 - CPU-Z

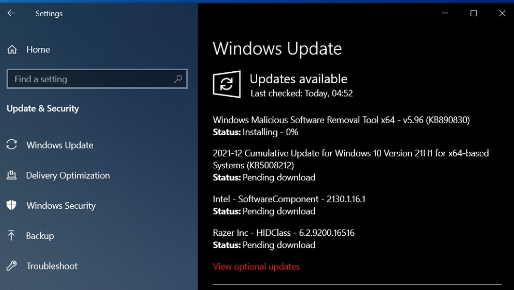


Рисунок 4 - Обновление системы Windows

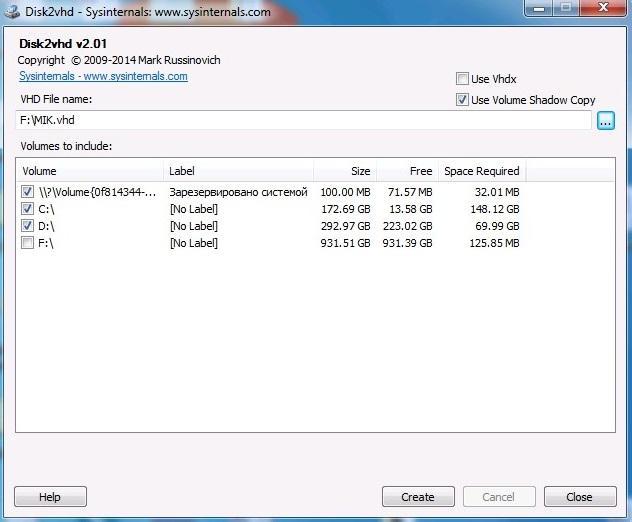


Рисунок 5 - Выбор настроек

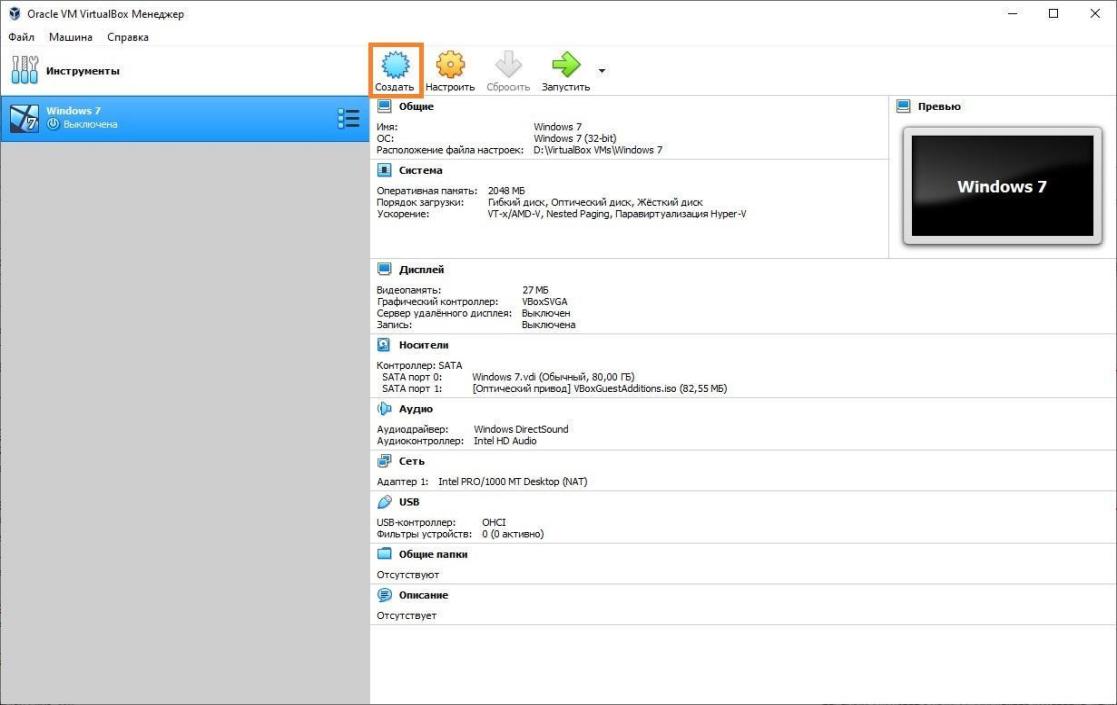


Рисунок 6 - Главное окно VirtualBox

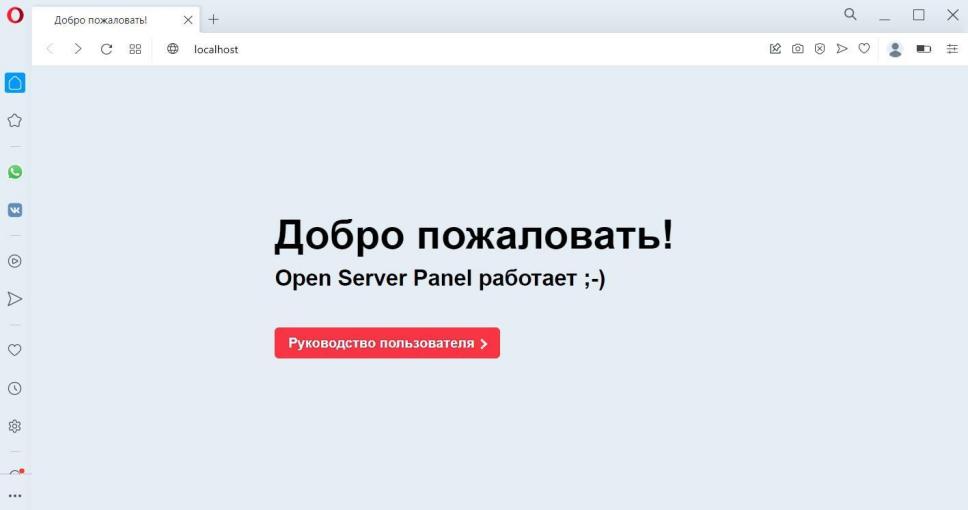


Рисунок 7 - Локальный хост

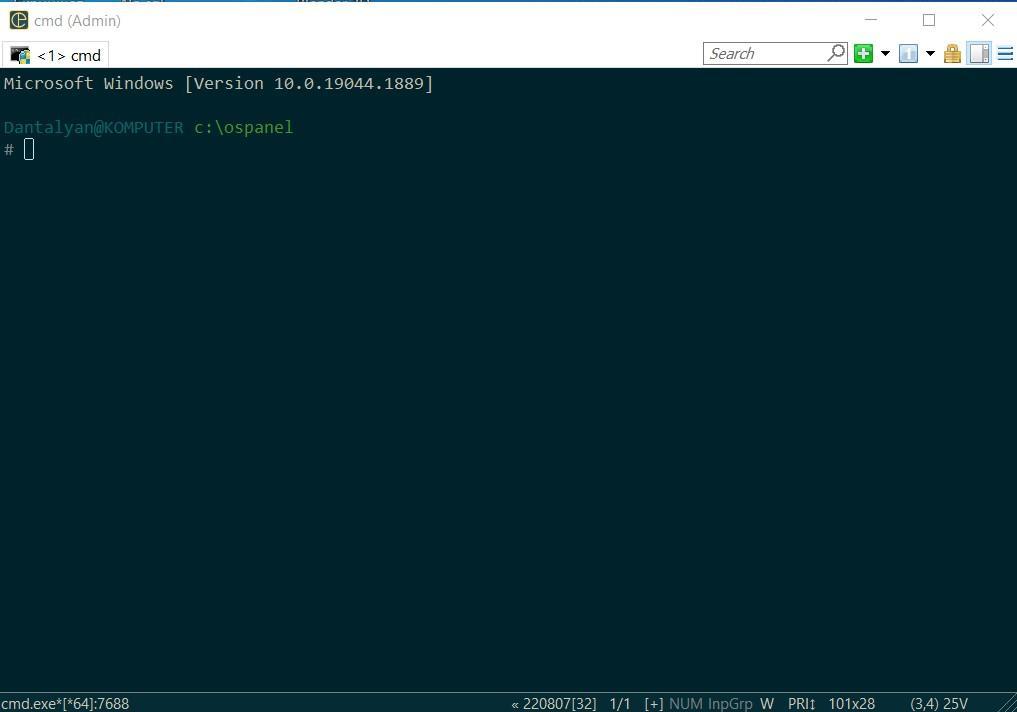


Рисунок 8 - Консоль сервера

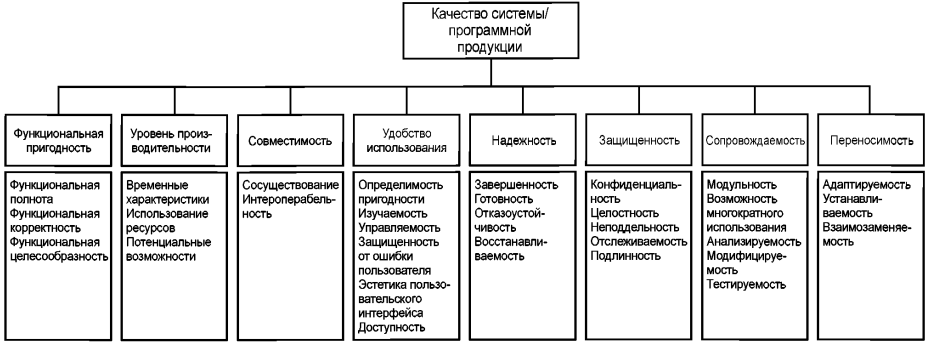


Рисунок 9 - Качество системы программной продукции

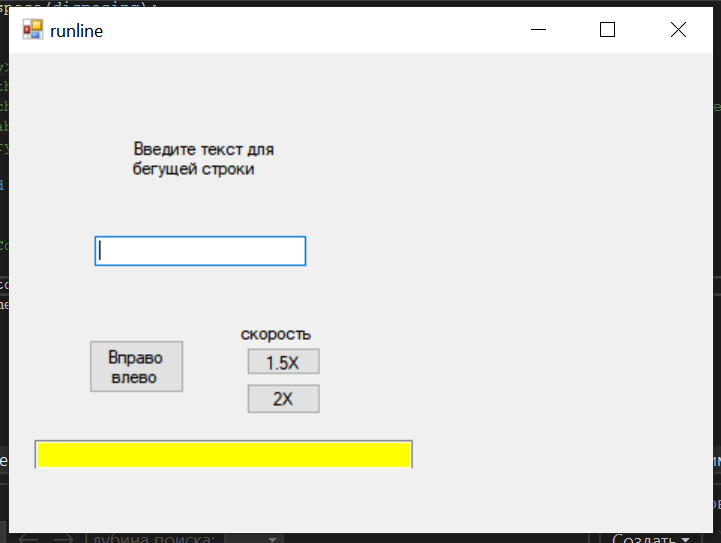


Рисунок 10 - Приложение с модулем

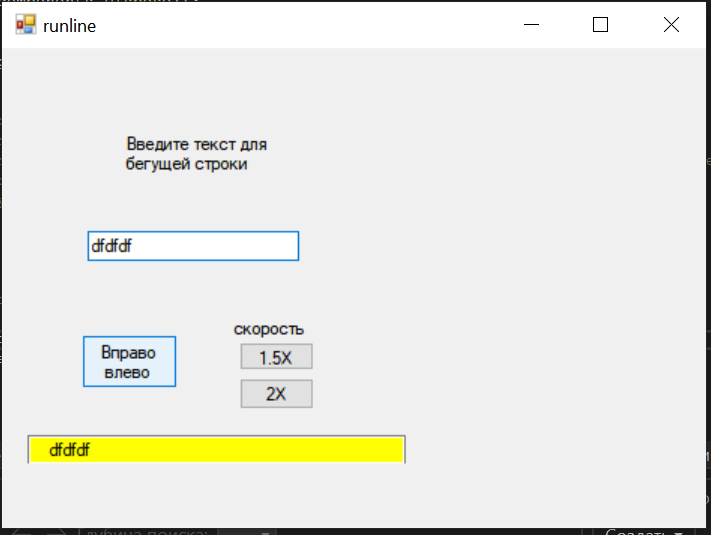


Рисунок 11 - Адаптация клиентского программного обеспечения

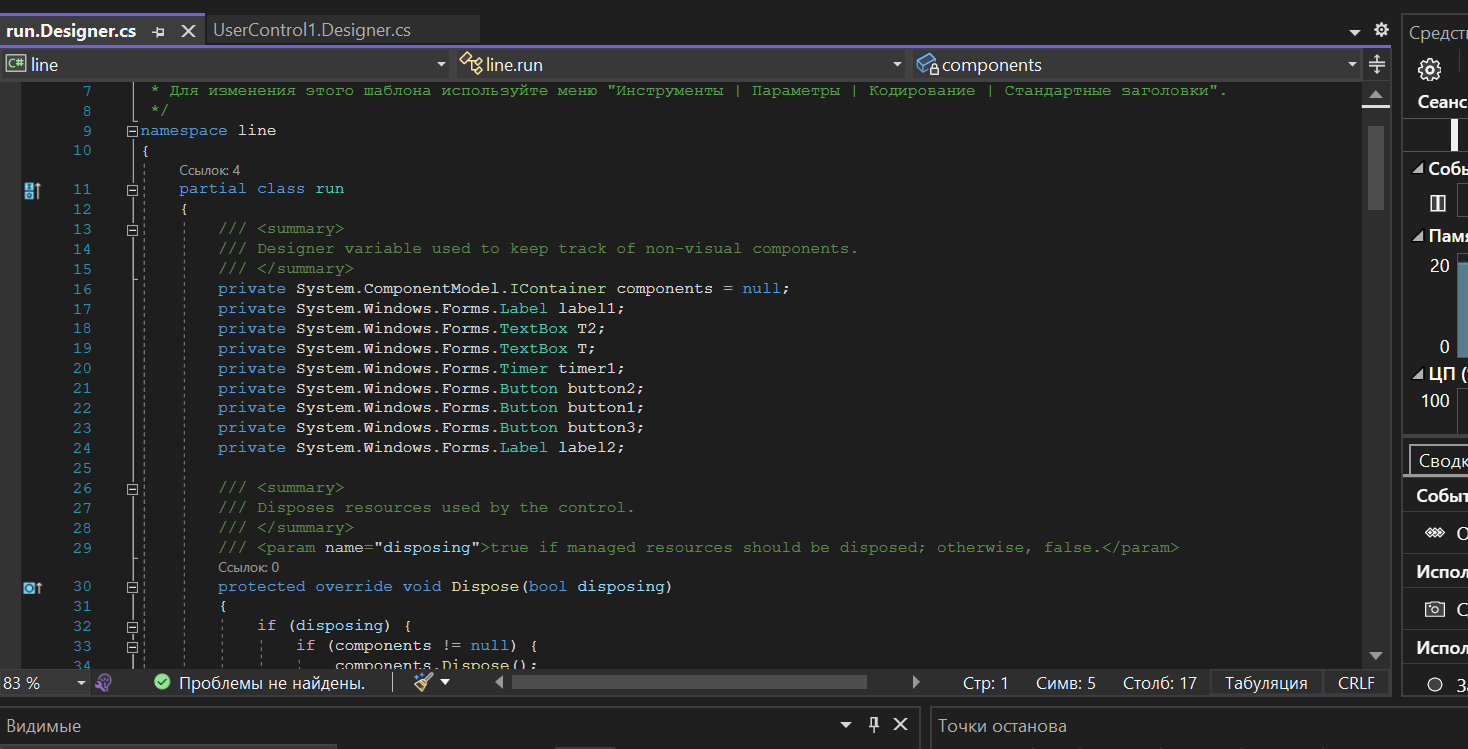


Рисунок 12 - Код модуля информационной системы

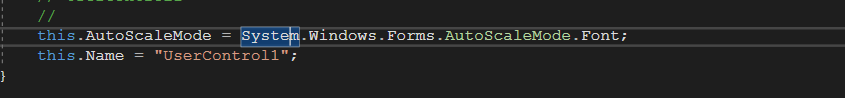


Рисунок 13 - Внедрение модуля в информационную систему

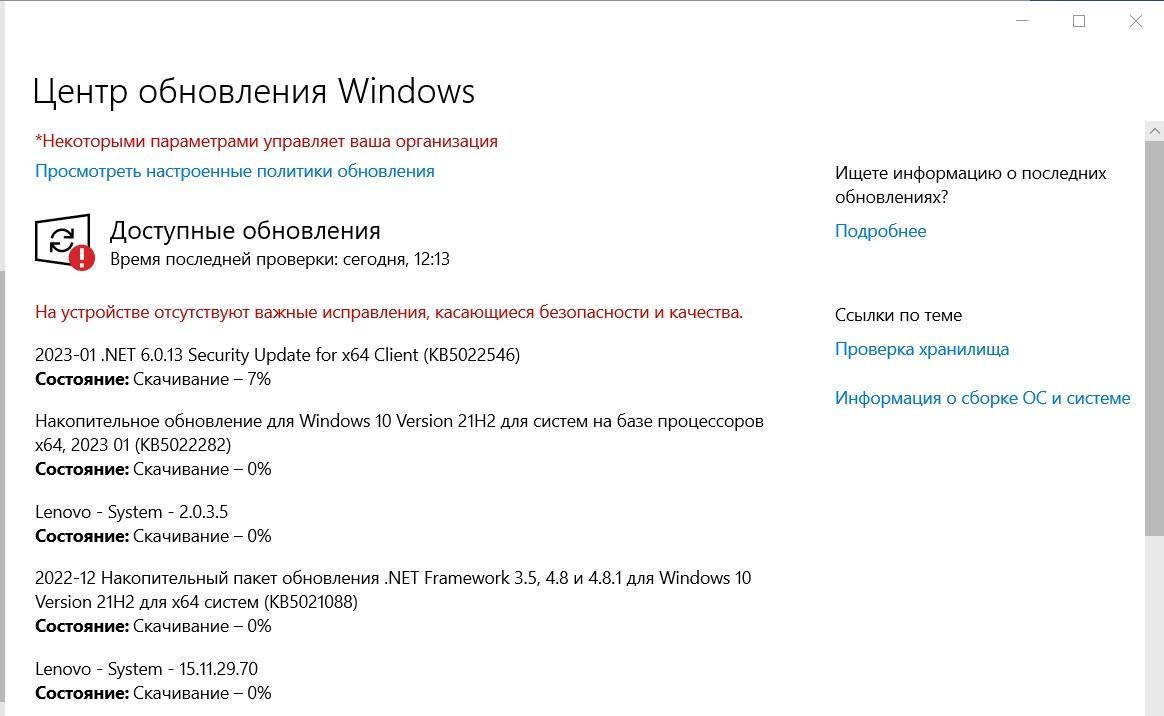


Рисунок 14 - Пример зеркала обновления

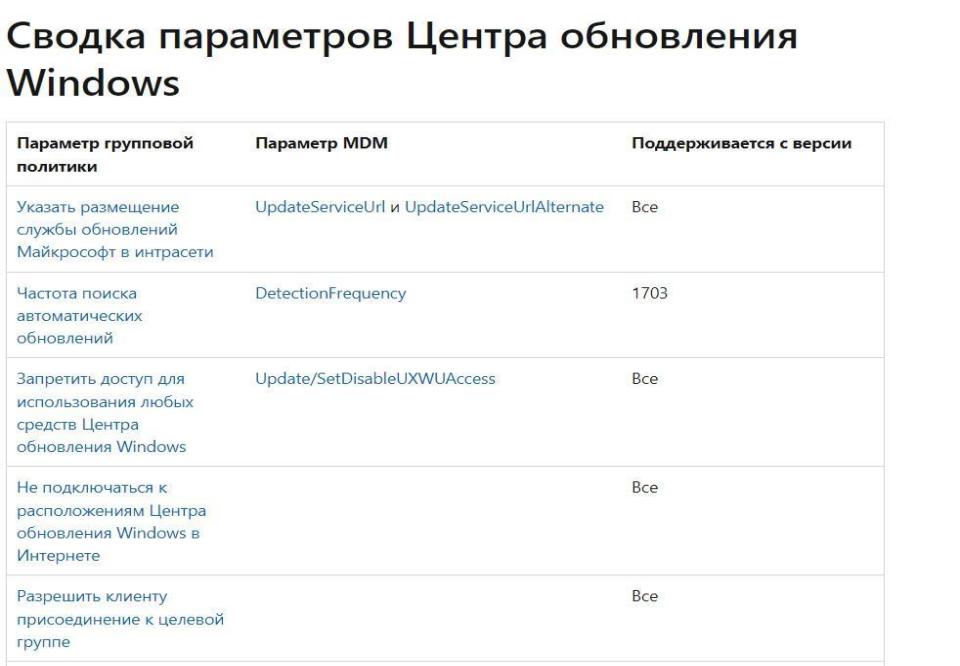


Рисунок 15 - Настройка обновлений

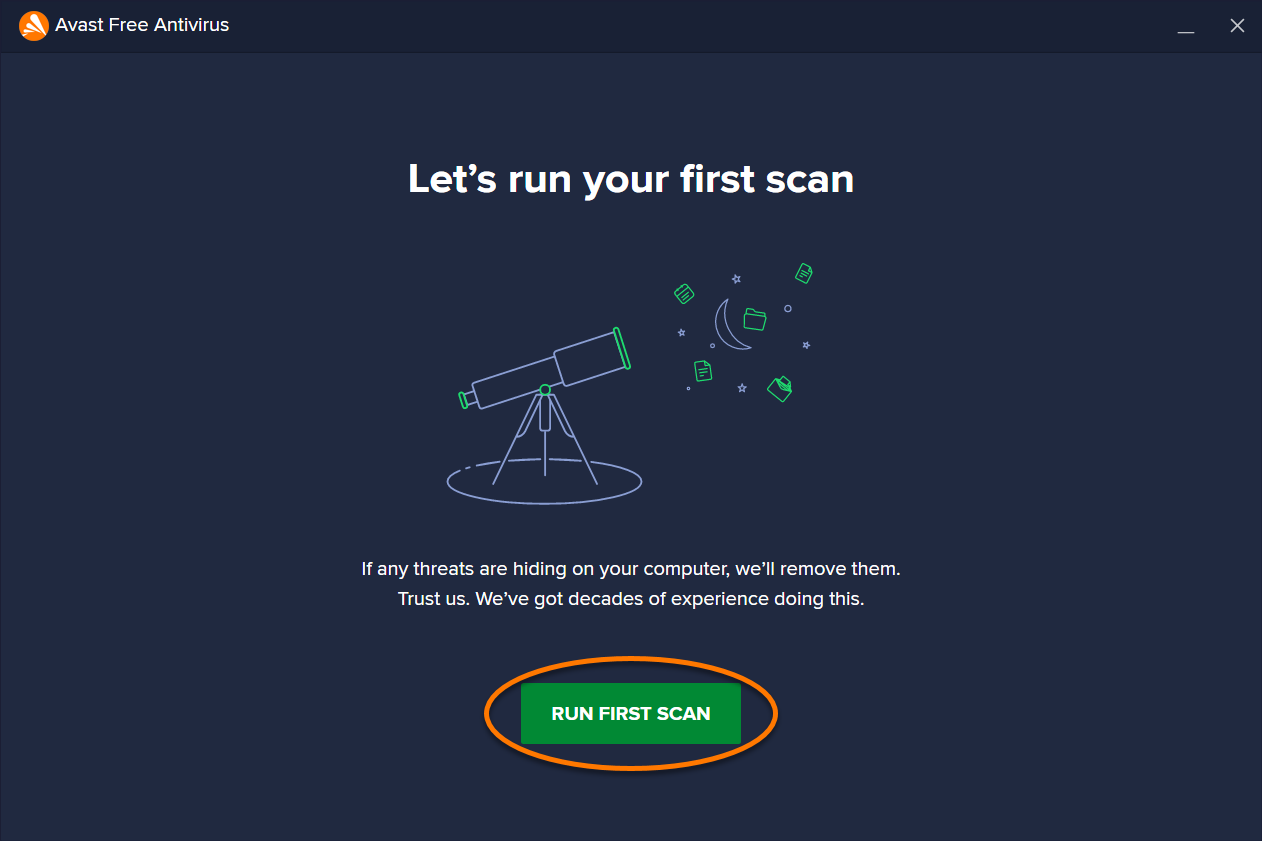


Рисунок 16 - Сканирование на вирусы

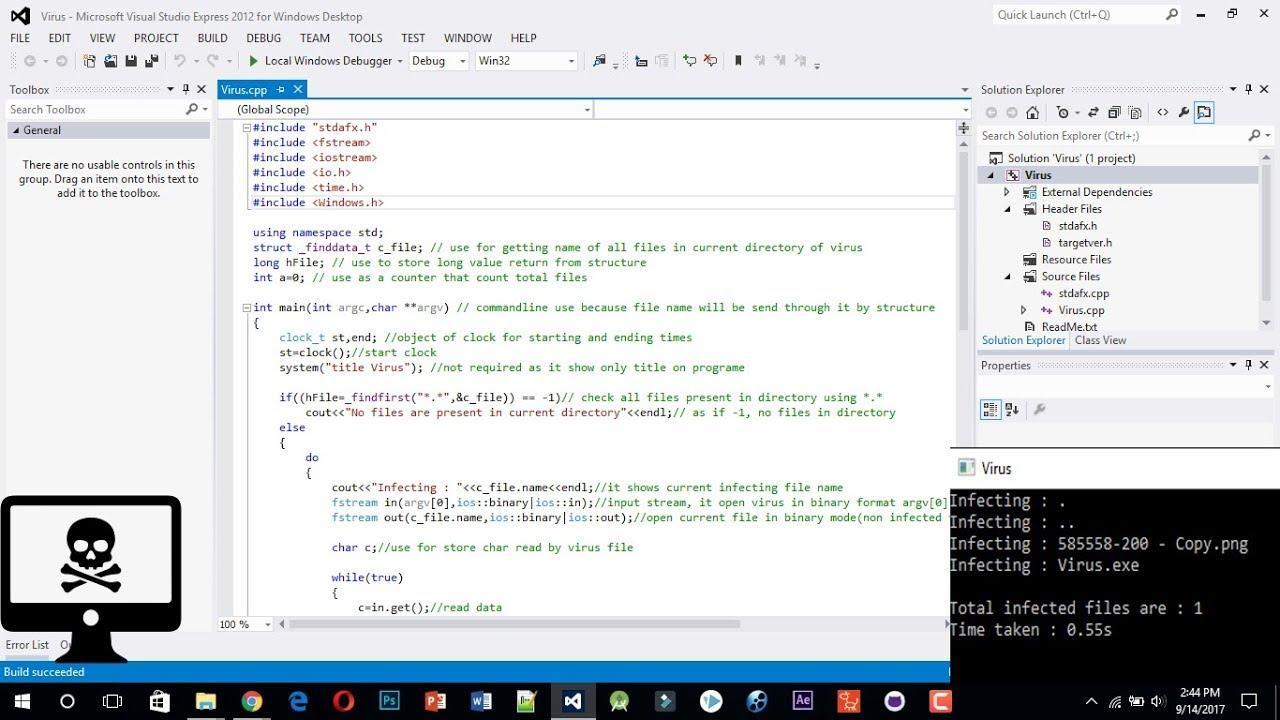


Рисунок 17 - Вирус, созданный на Visual Studio

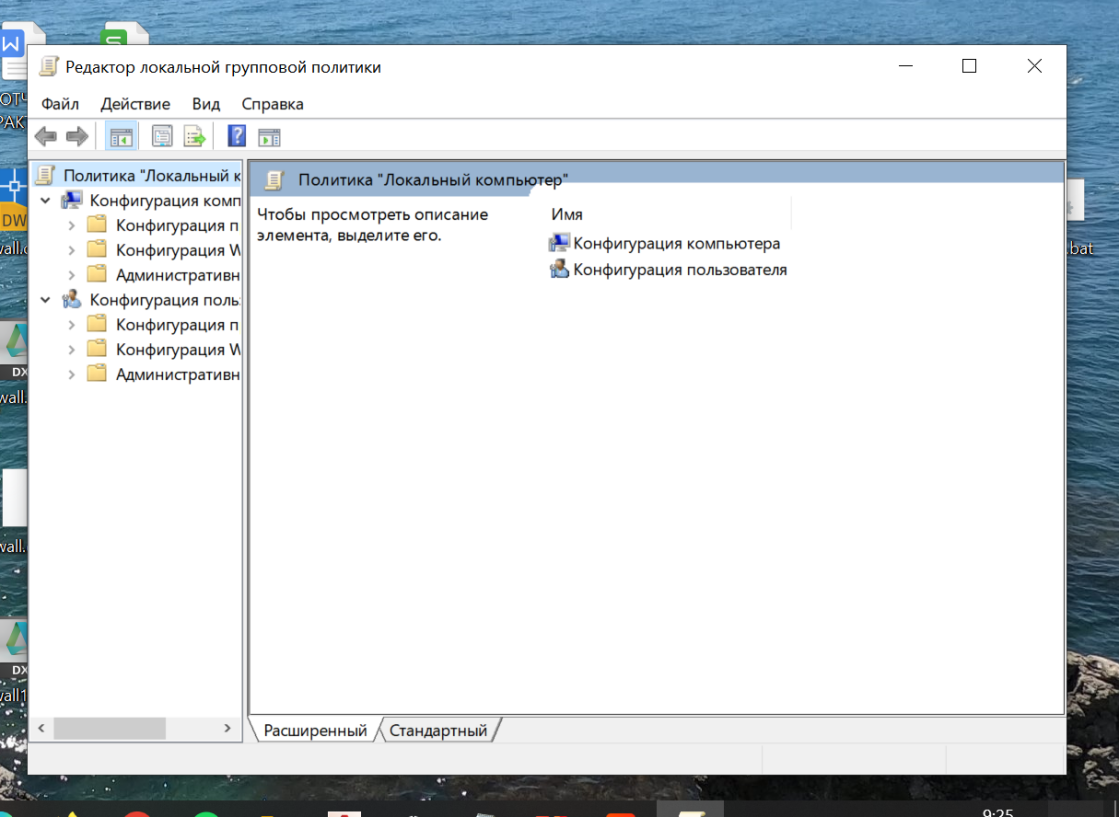


Рисунок 18 - Редактор политики безопасности

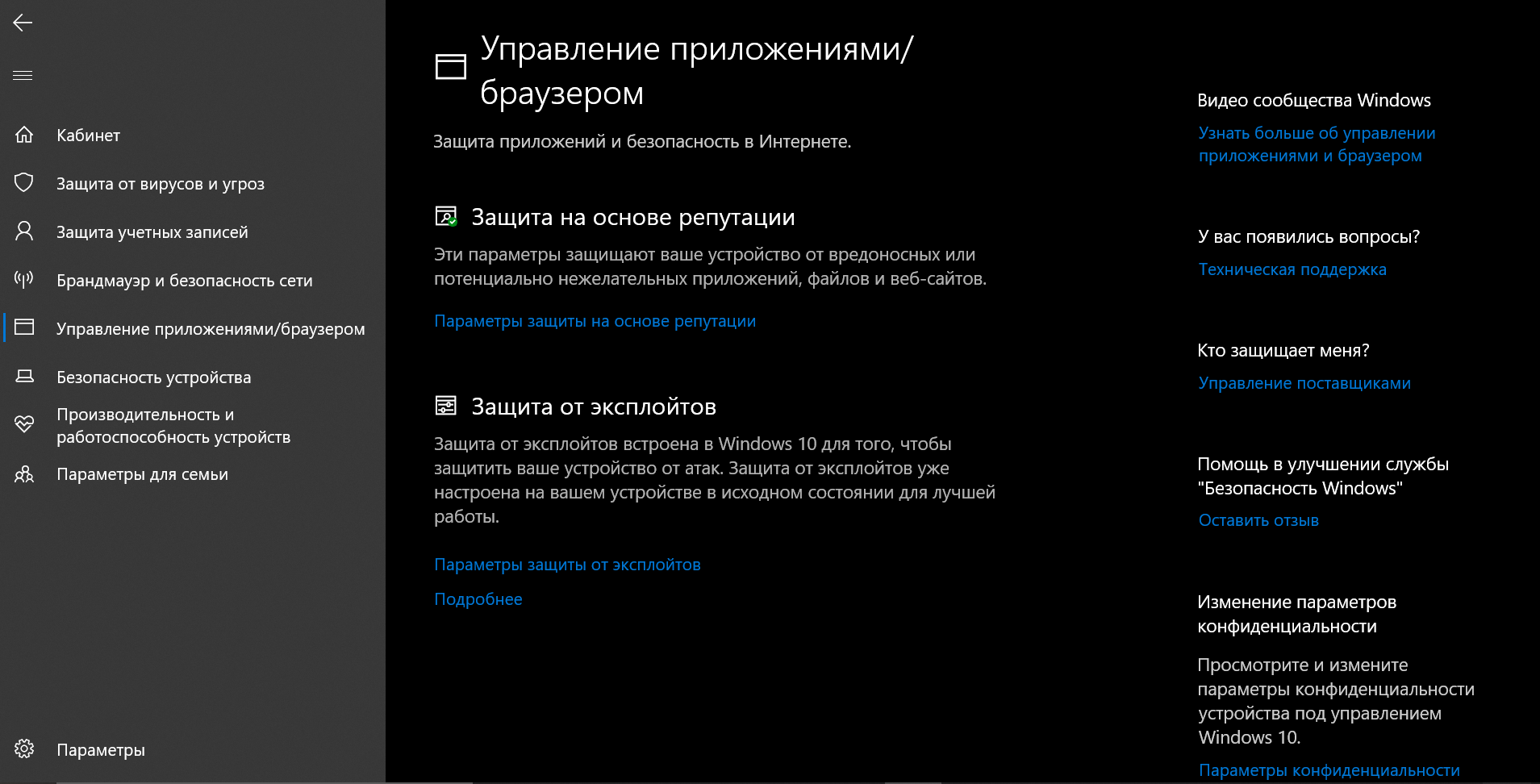


Рисунок 19 - Управление приложениями/браузером

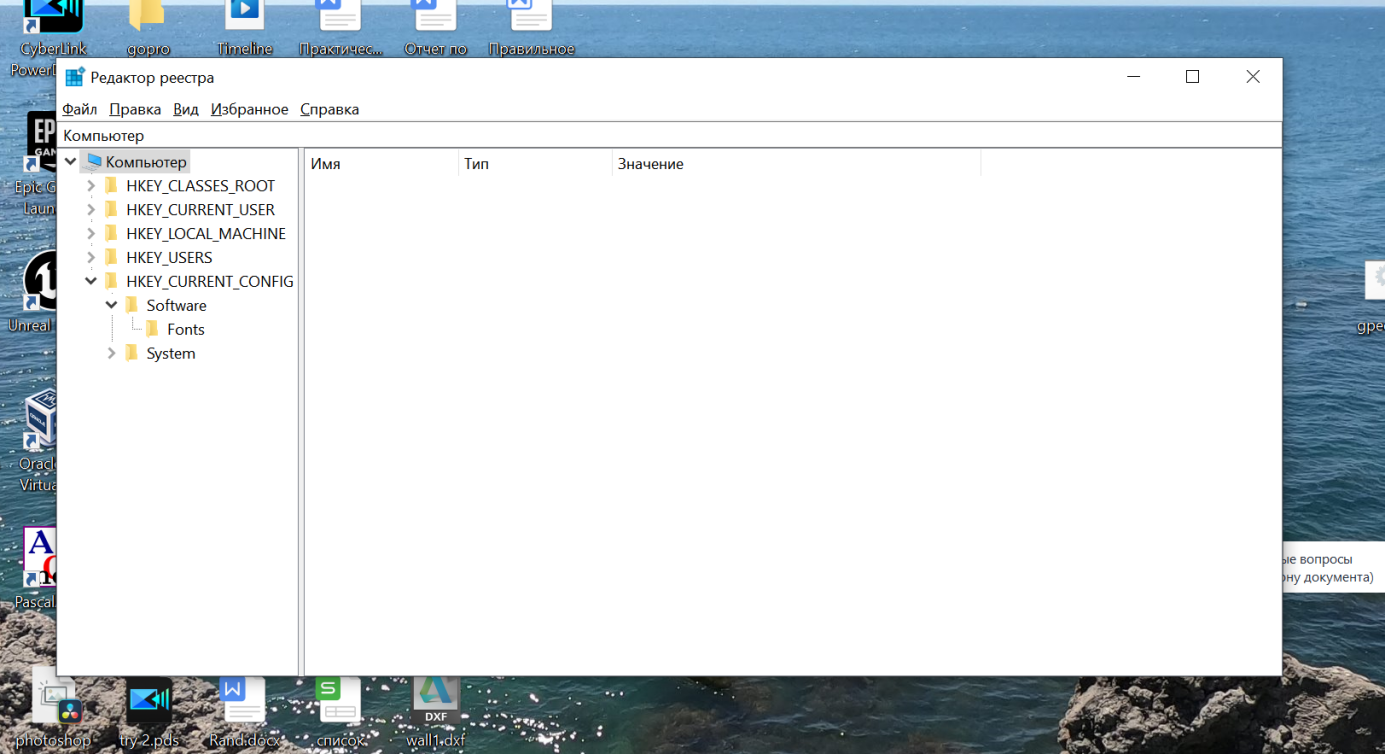


Рисунок 20 - Работа с реестром

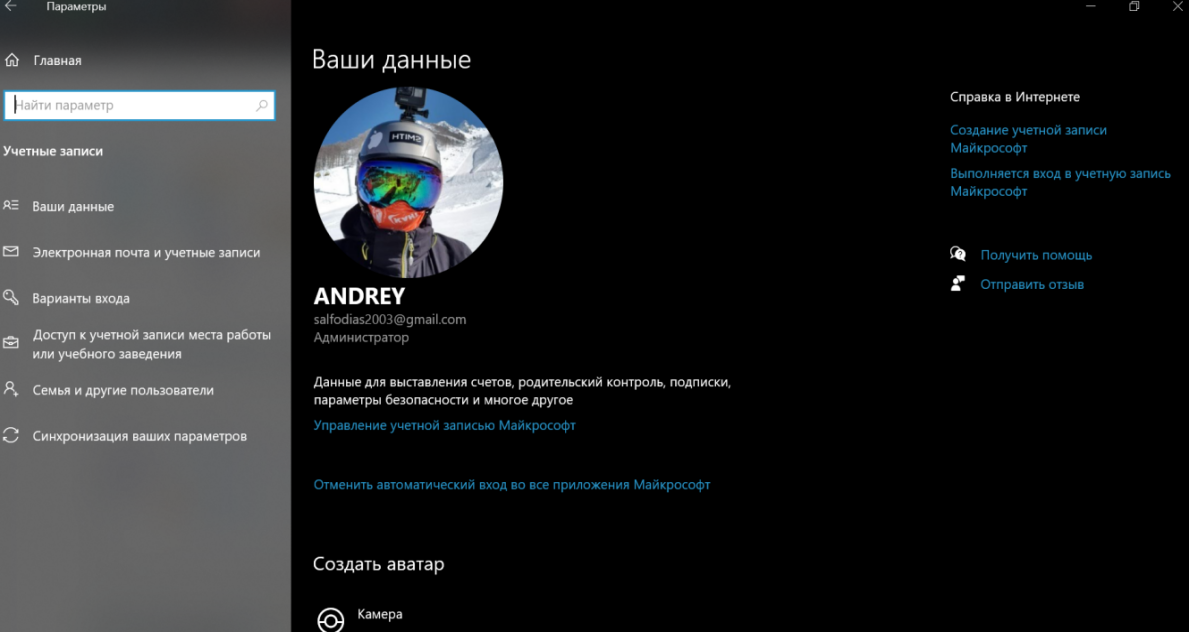


Рисунок 21 - Учетные записи

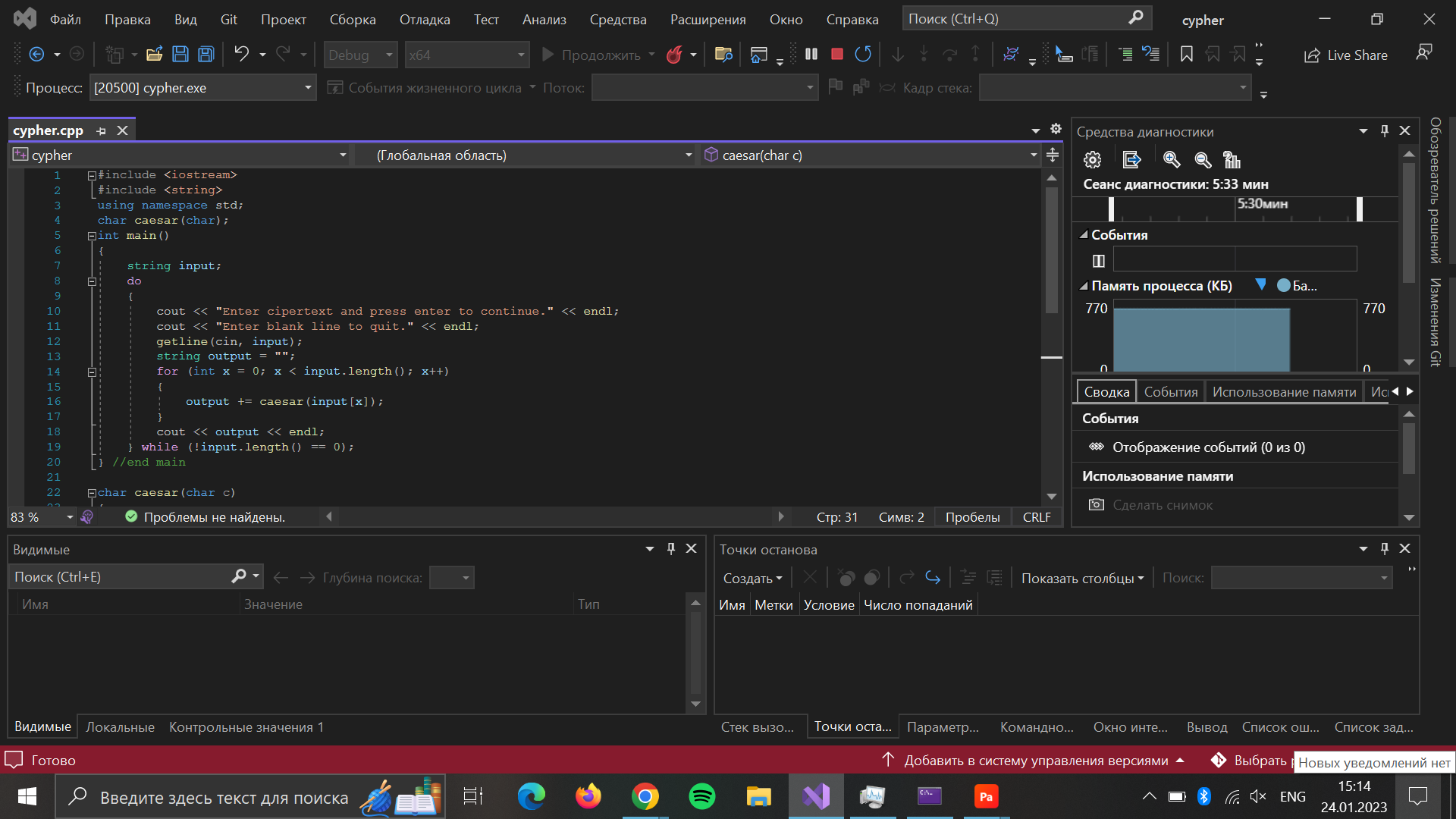


Рисунок 22 - Программа зашифровки и расшифровки

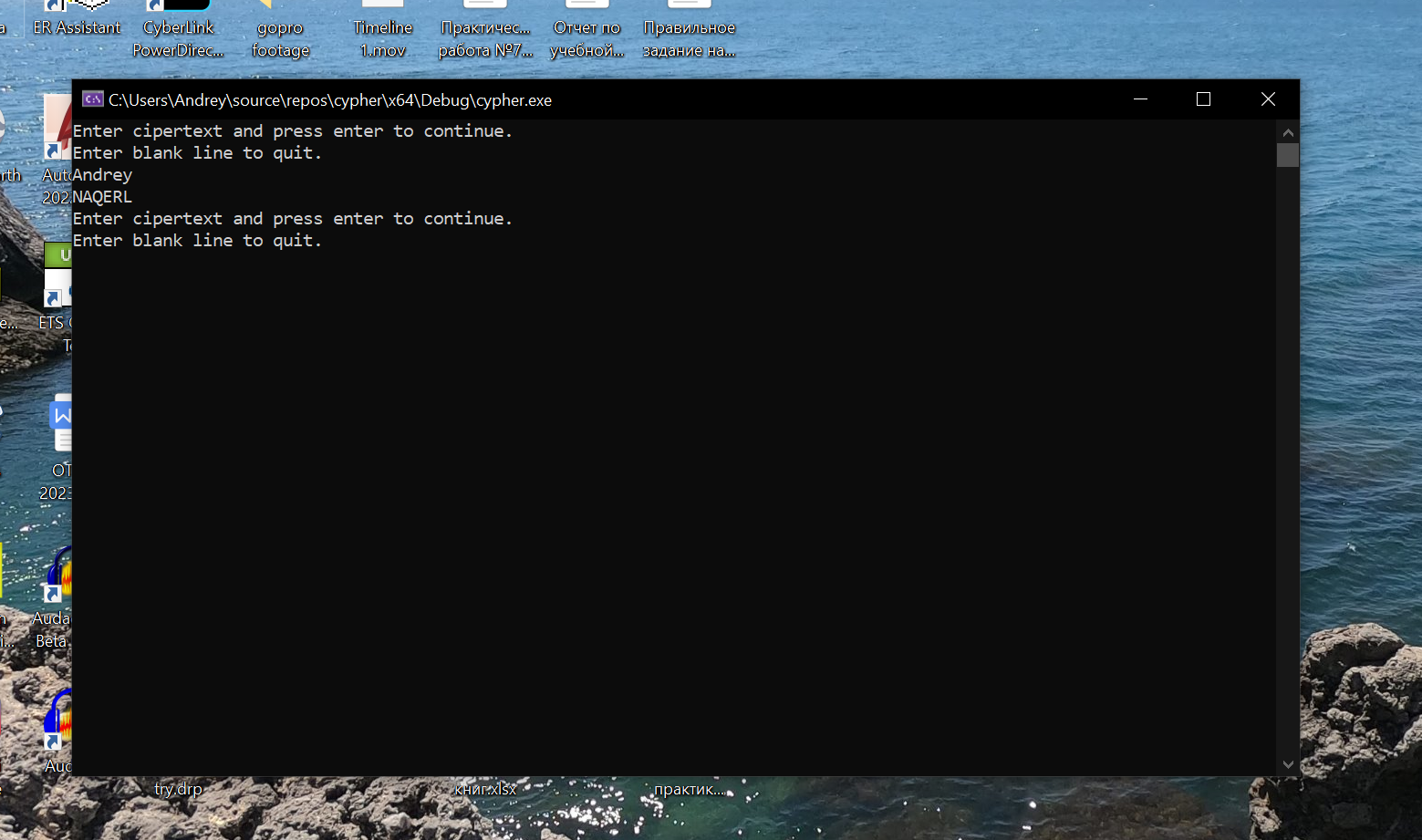


Рисунок 23 - Результат программы