**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ РЕСПУБЛИКИ**

**КЫРГЫЗСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ имени И.РАЗЗАКОВА**

**КАФЕДРА «ПРОГРАММНОЕ ОБЕСПЕСЕНИЕ**

**КОМПЬЮТЕРНЫХ СИСТЕМ»**

**ВЫПУСКНАЯ**

**КВАЛИФИКАЦИОННАЯ РАБОТА**

**Разработка клиент-серверного приложения автошколы и обеспечение её информационной безопасности**

на академическую степень бакалавра

**по направлению: 590100 «Информационная безопасность»**

|  |  |
| --- | --- |
| **Выполнил**  **Группа** | **Абыл уулу Нурзамат**  **ИБ(б)-1-20** |
| **Руководитель** | **доцент Вагнер Андрей Арнович** |

**Бишкек 2024**

**КЫРГЫЗ РЕСПУБЛИКАСЫНЫН БИЛИМ БЕРҮҮ**

**ЖАНА ИЛИМ МИНИСТРЛИГИ**

**И.РАЗЗАКОВ атындагы КЫРГЫЗ МАМЛЕКЕТТИК ТЕХНИКАЛЫК**

**УНИВЕРСИТЕТИ**

**«КОМПЬЮТЕРДИК СИСТЕМАЛАРДЫ ПРОГРАММАЛЫК КАМСЫЗДОО»**

**кафедрасы**

**БҮТҮРҮҮЧҮ**

**КВАЛИФИКАЦИЯЛЫК ИШ**

**Автомектептин клиент-сервердик тиркемесин иштеп чыгуу жана анын маалыматтык коопсуздугун камсыздоо**

бакалавр академиялык даражасы

**590100 «Маалыматтык коопсуздук» багыты боюнча**

|  |  |
| --- | --- |
| **Аткарган**  **Топ** | **Абыл уулу Нурзамат**  **ИБ(б)-1-20** |
| **Жетекчи** | **доцент Вагнер Андрей Арнович** |

**Бишкек 2024**

**АННОТАЦИЯ**

В данной выпускной квалификационной работе представлены результаты разработки клиент-серверного приложения автошколы и обеспечение ее информационной безопасности.

Данная работа состоит из 5 основных глав: анализ и разработка требований, конструкторские работы, проектирование безопасности, разработка документации, экспериментальный раздел. А также введение, заключение, список литературы, глоссария и приложения.

Глава «Анализ и разработка требований» описывает процессы клиент-серверного приложения автошколы, также в этой главе представлена классификация информации, идентификация активов.

В главе «Конструкторские работы» представлены разработка концептуальной модели и модели потоков данных клиент-серверного приложения.

В главе «Проектирование безопасности» представлена перечень процессов системы защиты информации, направленных на снижение уровня риска до приемлемого.

В главе «Разработка документации» содержится описание программы, а также руководство для администратора, программиста и пользователя.

В главе «Экспериментальный раздел» приводится план тестирования и результаты проведенного тестирования.

В состав данной работы также входят приложения.

Выпускная квалификационная работа состоит из 73 страниц, содержит 25 рисунков, количество таблиц 6, используется 9 источников.

**АННОТАЦИЯ**

Бул бүтүрүүчүлөрдүн квалификациялык ишинде автомектептин клиент-сервердик тиркемесин иштеп чыгуунун жана коопсуздугун камсыз кылуунун натыйжалары келтирилген.

Бул бүтүрүүчү иши 5 негизги бөлүмдөн турат: талаптарды изилдөө жана иштеп чыгаруу, конструктордук иштер, коопсуздукту долбоорлоо, документтештирүү иштери, эксперименталдык бөлүк. Ошондой эле кириш сөз, корутунду, колдонулган адабияттар, глоссарий жана тиркемеден турат.

«Талаптарды изилдөө жана иштеп чыгаруу» автомектептин клиент-сервердик тиркемесинин процесстерин сүрөттөйт, жана дагы бул бөлүмдө маалыматтын классификациясы жана активдерди идентификациялоо сунушталат.

«Конструктордук иштер» бөлүмүндө концептуалдык моделин жана агымдардын берилиши моделин иштеп чыгуусу чагылдырылган.

«Коопсуздукту долбоорлоо» бөлүмү тобокелдиктин деңгээлин алгылыктуу деңгээлге чейин төмөндөтүү үчүн багытталган маалыматтык коопсуздук тутумунун процесстеринин тизмесин камтыйт.

«Документтештирүү иштери» бөлүмүндө программанын баяндамасын, ошондой эле администратордун, програмисттин жана пайдалануучунун жетектемесин камтыйт.

«Эксперименталдык бөлүк» бөлүгүндө тестирлөөнүн планы жана тестирлөөнүн натыйжалары келтирилген.

Бул иш ошондой эле тиркемелерди камтыйт.

Бүтүрүүчүлүк квалификациялык иш 73 барактан турат, 25 сүрөттү камтыйт, жадыбалдардын саны 6, 9 булак колдонулган.

ОГЛАВЛЕНИЕ

[ПЕРЕЧЕНЬ УСЛОВНЫХ ОБОЗНАЧЕНИЙ. ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ 7](#_Toc168484297)

[ВВЕДЕНИЕ 8](#_Toc168484298)

[ЦЕЛЬ 10](#_Toc168484299)

[ЗАДАЧИ 10](#_Toc168484300)

[ГЛАВА 1. АНАЛИЗ И РАЗРАБОТКА ТРЕБОВАНИЙ 11](#_Toc168484301)

[**1.1** **Общая характеристика объекта защиты** 11](#_Toc168484302)

[**1.2** **Обоснование для разработки** 11](#_Toc168484303)

[**1.4** **Описание процессов** 11](#_Toc168484304)

[**1.5** **Функциональные требования** 12](#_Toc168484305)

[**1.6** **Основные процессы и информационные потоки** 12](#_Toc168484306)

[**1.7** **Требования к составу и параметрам технических средств** 13](#_Toc168484307)

[**Требования к системе** 13](#_Toc168484308)

[**1.4** **Требования к безопасности** 13](#_Toc168484309)

[**1.5** **Требования к средствам и способам связи для информационного обмена между компонентами системы.** 14](#_Toc168484310)

[**1.6** **Требования по диагностированию системы** 15](#_Toc168484311)

[**1.7** **Требования к надежности комплекса.** 15](#_Toc168484312)

[ГЛАВА 2. КОНСТРУКТОРСКИЕ РАБОТЫ 16](#_Toc168484313)

[**2.1** **Декомпозиция концептуальной модели** 16](#_Toc168484314)

[**2.2** **Организация структуры данных** 18](#_Toc168484315)

[**2.3** **Архитектура разрабатываемой системы** 19](#_Toc168484316)

[ГЛАВА 3. ПРОЕКТИРОВАНИЕ БЕЗОПАСНОСТИ 21](#_Toc168484317)

[**3.1 Процессы организации** 21](#_Toc168484318)

[**3.2 Инфраструктура клиент-серверного приложения** 21](#_Toc168484319)

[**3.3 Идентификация активов** 21](#_Toc168484320)

[**3.4 Модель нарушителя** 22](#_Toc168484321)

[**3.5 Модель угроз** 23](#_Toc168484322)

[**3.6** **Способы защиты от вероятных атак** 27](#_Toc168484323)

[**3.6.1 Log4Qt - Логирование** 28](#_Toc168484324)

[**3.6.2 TinyWall - FireWall** 29](#_Toc168484325)

[**3.6.3 Модуль QT-Secret** 30](#_Toc168484326)

[ГЛАВА 4. РАЗРАБОТКА ДОКУМЕНТАЦИИ 33](#_Toc168484327)

[**4.1** **Описание программы** 33](#_Toc168484328)

[**4.2** **Руководство разработчика** 33](#_Toc168484329)

[ГЛАВА 5. ТЕСТИРОВАНИЕ 35](#_Toc168484330)

[**5.1** **DoS атака** 35](#_Toc168484331)

[**5.2** **SQL инъекция** 36](#_Toc168484332)

[ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ 38](#_Toc168484333)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ 39](#_Toc168484334)

[ПРИЛОЖЕНИЯ 41](#_Toc168484335)

[ПРИЛОЖЕНИЕ А 42](#_Toc168484336)

[ПРИЛОЖЕНИЕ Б 45](#_Toc168484337)

[ПРИЛОЖЕНИЕ В 50](#_Toc168484338)

# **ПЕРЕЧЕНЬ УСЛОВНЫХ ОБОЗНАЧЕНИЙ. ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ**

В данной выпускной квалификационной работе применяются следующие термины с соответствующими определениями:

Актив – это все, что имеет ценность для организации.

ВВЕДЕНИЕ

В современном мире, где информационные технологии стали неотъемлемой частью многих сфер деятельности, включая образование, многие учебные организации прибегают к использованию клиент-серверных приложений. В частности, автошколы активно внедряют технологии для улучшения своей эффективности и качества обучения. Тем самым разработка и обеспечение безопасности клиент-серверного приложения автошколы становится критически значимым аспектом, требующим к себе особого внимания.

Учитывая конфиденциальные информации, такие как данные курсантов и инструкторов, необходимо предпринимать шаги для защиты от утечек и несанкционированного доступа. Это включает в себя шифрование данных, управление доступом, регулярное обновление системы безопасности и мониторинг активности для выявления подозрительных ситуаций.

Также одним из ключевых аспектов разработки такого приложения является создание интуитивно понятного интерфейса как для инструкторов, так и для курсантов. Удобство использования приложения включает в себя функциональность для записи на занятия, отслеживания прогресса, доступа к учебным материалам и другим аспектам обучения.

Кроме того, важно обеспечить стабильную работу приложения, чтобы избежать проблем в процессе обучения. Регулярные тестирования, обновления и поддержка приложения помогут устранить возможные сбои и повысить его надежность.

Таким образом, разработка и обеспечение безопасности клиент-серверного приложения автошколы требует комплексного подхода, включающего в себя удобство использования, защиту данных и стабильность работы..

АКТУАЛЬНОСТЬ РАБОТЫ

В ходе выполнения анализа автошколы, были выявлены следующие проблемы:

***1. Неэффективное управление рутинными задачами:***

* Ручное ведение журнала посещаемости, регистрации курсантов, планирования занятий,
* Длительное время обработки заявок на обучение.
* Ошибки в документации.
* Неоптимальное использование ресурсов.

***2. Низкое качество обучения:***

* Отсутствие доступа к обучающим материалам 24/7.
* Неперсонализированный подход к обучению.
* Неэффективные методы обучения.
* Недостаточная обратная связь.

***3. Снижение конкурентоспособности:***

* Отсутствие современных технологий.
* Неудовлетворенность клиентов.
* Ограниченный спектр услуг.

***4. Повышенные риски:***

* Ошибки, связанные с человеческим фактором.
* Несоответствие требованиям законодательства.

В связи с этими проблемами, было принято решение разработать и обеспечить безопасность клиент-серверного приложения для автошколы.

ЦЕЛЬ

Цель работы заключается в разработке и обеспечении информационной безопасности клиент-серверного приложения автошколы, включающая в себя состав:

* Управление базой данных курсантов;
* Работу сотрудников;
* Обеспечение безопасности системы и всех основных процессов, входящих в состав автошколы;

ЗАДАЧИ

1. Анализ предметной области.
2. Обзор системы организации.
3. Описание бизнес процессов.
4. Описание связи между бизнес процессами.
5. Разработка модели угроз.
6. Разработка модели нарушителя.

**ГЛАВА 1. АНАЛИЗ И РАЗРАБОТКА ТРЕБОВАНИЙ**

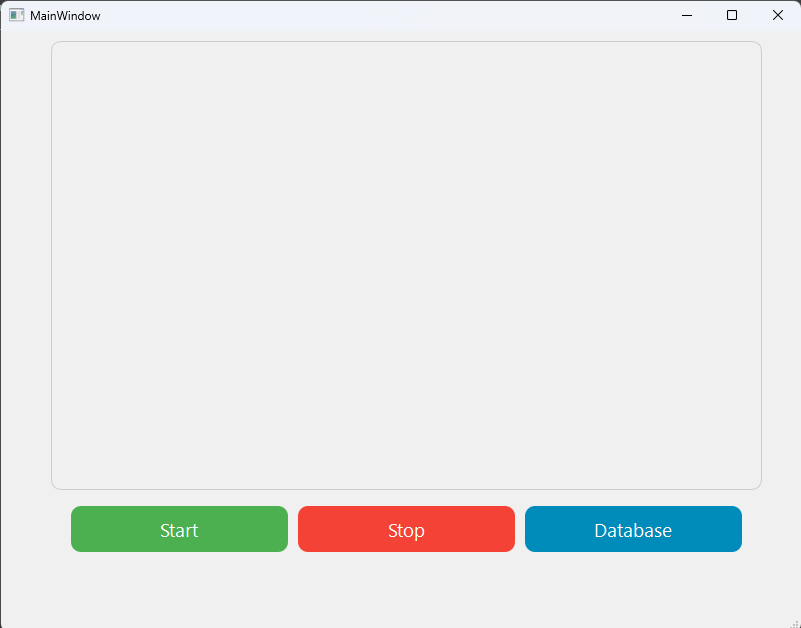
* 1. **Общая характеристика объекта защиты**

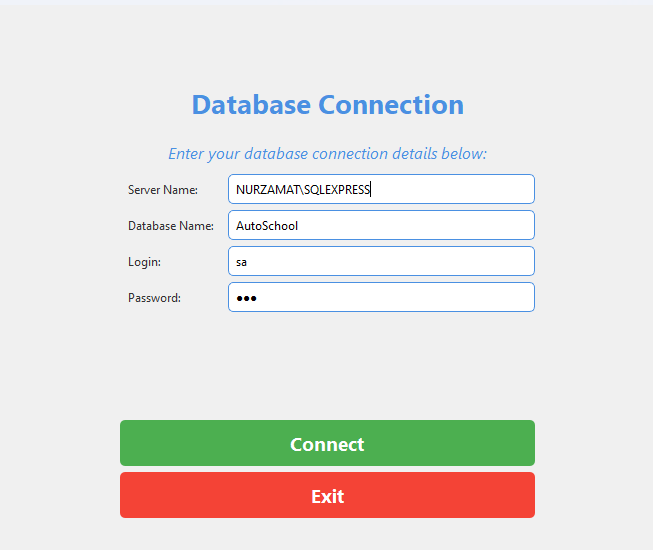
Клиентское приложение разработано на фреймворке QT v6 языка С++. Хранение базы данных производится на СУБД SQL Server. Целью клиент-серверного приложения является оптимизация работы и улучшение взаимодействия преподавателей и курсантов. Внедрение является практической мерой по обеспечению эффективности производственного процесса. Для работы приложения используется сервер написанный также на фреймворке QT.

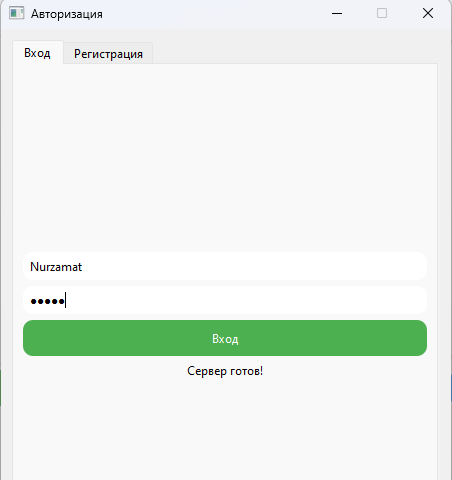
* 1. **Обоснование для разработки**

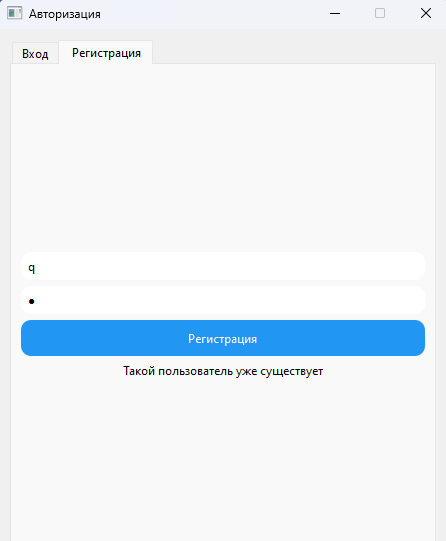
Разработанное клиент-серверное приложение для автошколы призвано для внутри организационного пользования, с хранением и обработкой внутри себя персональных данных сотрудников и курсантов. Хранение персональных данных внутри приложения облегчает их доступность и конфиденциальность, а также исключает возможность потери или порчи в сравнении с хранением бумажных версий в архивах.

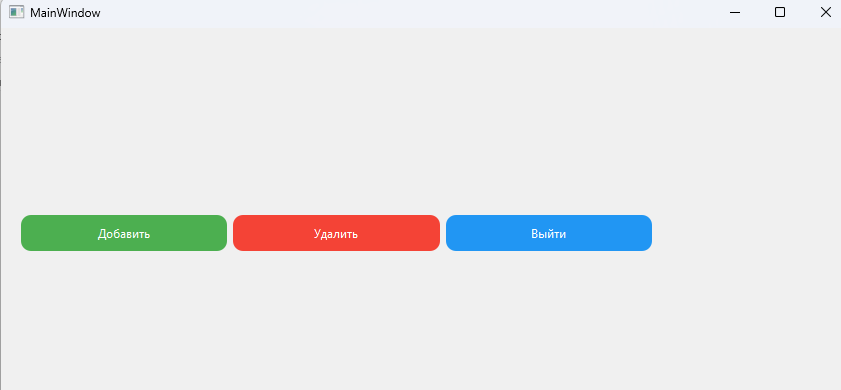
* 1. **Обзор системы автошколы**

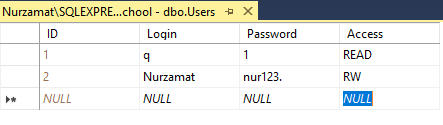


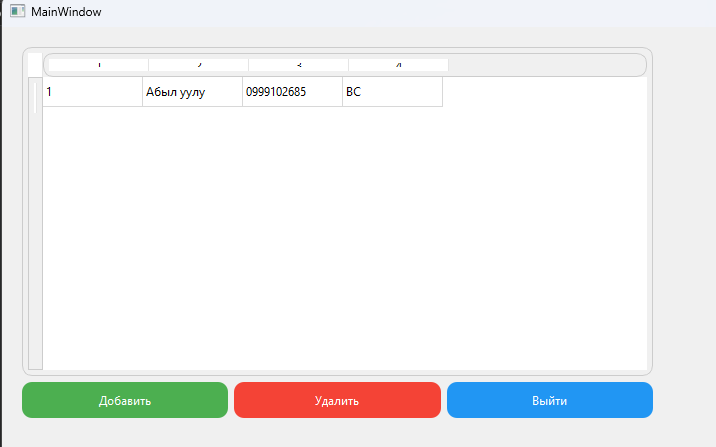


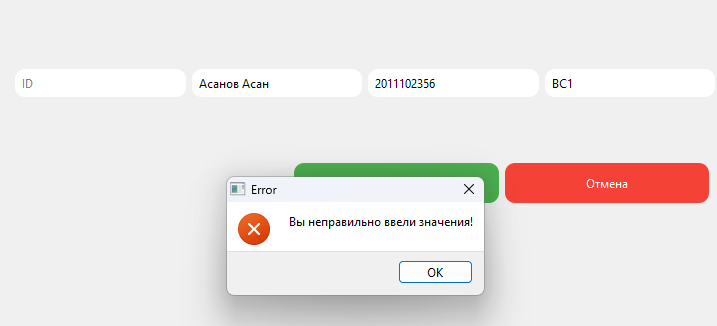


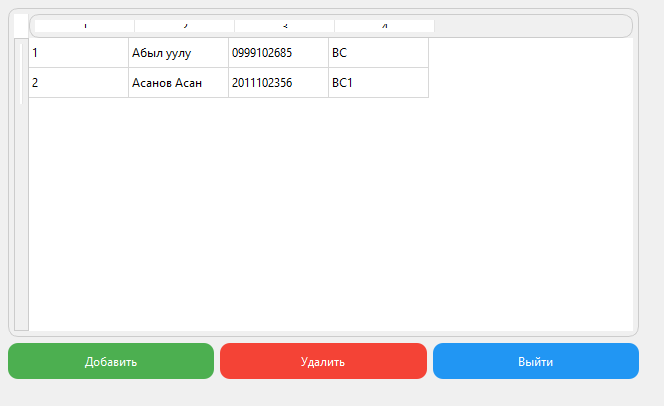












* 1. **Описание процессов**

Таблица 1.1. Процессы

|  |  |  |
| --- | --- | --- |
| **Процесс** | **Актив** | **Среда обработки** |
| Авторизация пользователей | Персональные данные | ПК, Сервер |
| Регистрация пользователей | Персональные данные | ПК, Сервер |
| Добавить группу | Информация ограниченного доступа | ПК, Сервер |
| Добавить курсанта | Информация ограниченного доступа | ПК, Сервер |
| Добавить расписание | Открытый доступ | ПК, Сервер |
| Добавить успеваемости курсантов | Информация ограниченного доступа | ПК, Сервер |
| Справочник по сотрудникам, курсантам | Информация ограниченного доступа | ПК, Сервер |

* 1. **Функциональные требования**

Разрабатываемое клиент-серверное приложение должно выполнять следующие функции:

* Регистрация / Авторизация
* Просмотр справочника сотрудников, курсантов
* Скачивание отчетов
* Введение успеваемости курсантов
* Введение расписания занятий
* Добавление, удаление, обновление групп, курсантов, сотрудников
* Нагрузка преподавателей
  1. **Основные процессы и информационные потоки**

В процессе проведения анализа требований к разрабатываемой системе, была создана модель в виде диаграммы IDEF0 (рис. 1.1.). Данная диаграмма показывает процессы, которые протекают внутри автошколы в ходе выполнения работ, что позволяет более детально рассмотреть проблемы и определить уязвимые места системы.

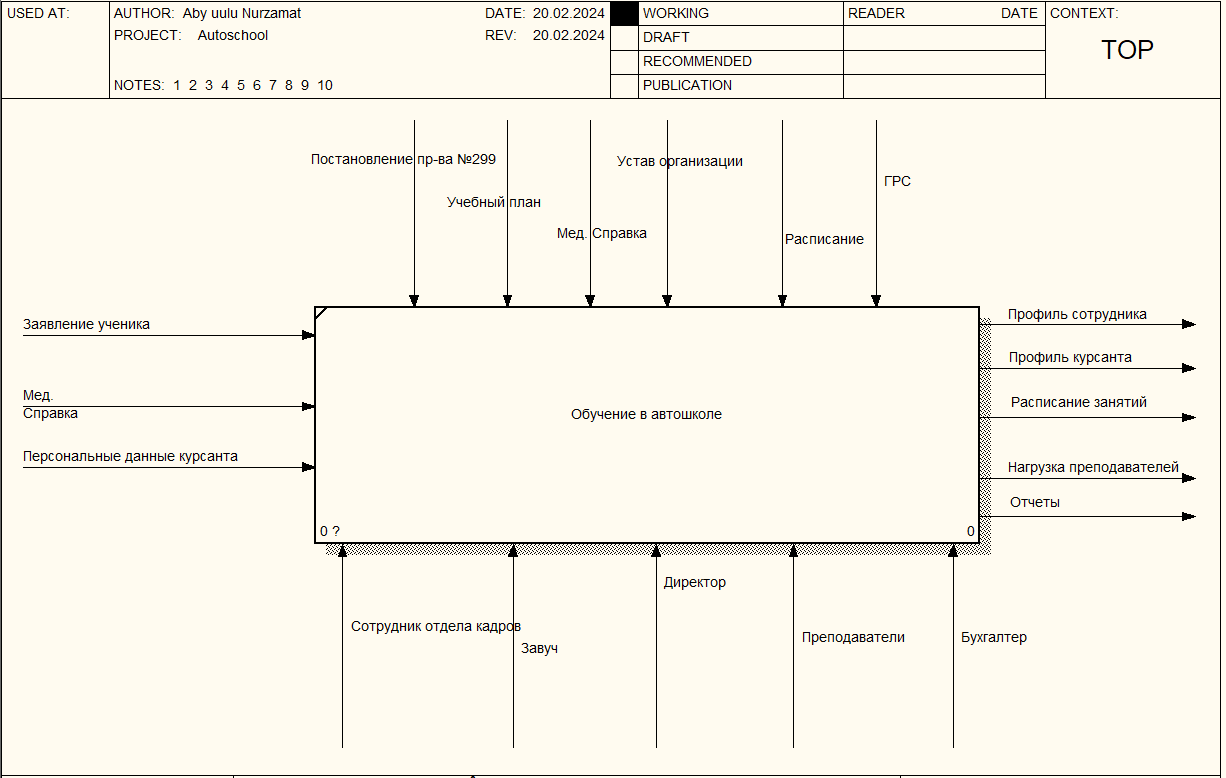
****

Рис. 1.1. Контекстная диаграмма основных процессов автошколы

* 1. **Требования к составу и параметрам технических средств**

Требования к аппаратному обеспечению:

• Тип процессора:

- Процессор x64: AMD Opteron, AMD Athlon 64, Intel Xeon с поддержкой Intel EM64T, Intel.

Pentium IV с поддержкой EM64T.

- Процессор x86: процессор, совместимый с Pentium III или выше.

• Быстродействие процессора: минимум 1,4 ГГц. Рекомендуется 2 ГГц и выше.

• ОЗУ: для обеспечения оптимальной производительности требуется не менее 4 ГБ с последующим увеличением по мере роста размера базы данных.

• Свободного дискового пространства для базы данных – 2 ГБ, так как со временем размер базы данных будет увеличиваться.

## **Требования к системе**

* 1. **Требования к безопасности**

**Подсистема (шифрование данных):**

**Надежность:** Алгоритм шифрования должен быть надежным и стойким к взлому.

**Соответствие стандартам:** Алгоритм шифрования должен соответствовать международным стандартам, таким как AES, ГОСТ 28147-89, RSA.

**Скорость:** Алгоритм шифрования должен работать достаточно быстро, чтобы не влиять на производительность системы.

**Длина ключа:** Длина ключа должна быть достаточно большой, чтобы обеспечить надежную защиту данных.

**Подсистема (идентификация, аутентификация, авторизация):**

**Соответствие стандартам:** Система идентификации должна соответствовать международным стандартам, таким как ISO/IEC 27001, NIST SP 800-63.

**Многофакторная аутентификация:** Должна быть предусмотрена возможность использования многофакторной аутентификации.

**Подсистема (управление доступом):**

**Соответствие стандартам:** Система управления доступом должна соответствовать международным стандартам, таким как ISO/IEC 27001, NIST SP 800-53.

**Гибкость:** Система управления доступом должна быть гибкой и способной адаптироваться к меняющимся потребностям вашей организации.

Обеспечение информационное безопасности Системы должно удовлетворять следующим требованиям:

• Защита Системы должна обеспечиваться комплексом программно-технических средств и поддерживающих их организационных мер.

• Защита Системы должна обеспечиваться на всех технологических этапах обработки информации и во всех режимах функционирования, в том числе при проведении ремонтных и регламентных работ.

• Программно-технические средства защиты не должны существенно ухудшать основные функциональные характеристики Системы (надежность, быстродействие, возможность изменения конфигурации).

• Разграничение прав доступа пользователей и администраторов Системы должно строиться по принципу, что не разрешено, то запрещено и т.д.

• Система должна обеспечивать обработку конфиденциальной информации.

• Применяемые в системе средства и технологии защиты должны обеспечивать открытость архитектуры и обладать свойствами модульности, масштабируемости и возможности адаптации системы к различным организационным и техническим условиям.

• Система должна обеспечивать необходимую и достаточную защиту ресурсов от характерных угроз безопасности, определенных с учетом объективных факторов и анализа возможных моделей нарушителей.

• Средства защиты, входящие в состав системы, должны иметь развитые средства регистрации критических системных событий в электронных журналах и средства оперативного оповещения об этих событиях администраторов безопасности.

* 1. **Требования к средствам и способам связи для информационного обмена между компонентами системы.**

Для информационного обмена между компонентами системы должна быть организована глобальная сеть.

* Использование стандартных протоколов связи, таких как HTTP/HTTPS, SMTP, FTP/SFTP, LDAP, для обеспечения совместимости и безопасности.
  1. **Требования по диагностированию системы**

Диагностика и профилактика технических средств, проводится раз в месяц. Проверка целостности данных и нарушений проводится по мере необходимости. Проверка программного и аппаратного обеспечения проводится по мере необходимости.

* 1. **Требования к надежности комплекса.**

Надежность аппаратных и программных средств должно обеспечиваться за счет следующих организационных мероприятий:

* использованием в системе технических средств повышенной отказоустойчивости и их структурным резервированием
* защитой технических средств по электропитанию путем использования источников бесперебойного питания
* предварительного обучения пользователей и обслуживающего персонала
* своевременного выполнения процессов администрирования
* соблюдения правил эксплуатации и технического обслуживания программно-аппаратных средств
* своевременное выполнение процедур резервного копирования данных

Надежность программного обеспечения системы должна обеспечиваться за счет:

* проведением комплекса мероприятий отладки, поиска и исключения ошибок
* ведением журналов системных сообщений и ошибок для последующего анализа и изменения конфигурации.

# **ГЛАВА 2. КОНСТРУКТОРСКИЕ РАБОТЫ**

## **Декомпозиция концептуальной модели**

На рисунке 2.1 и 2.2 представлены декомпозиции контекстной диаграммы.

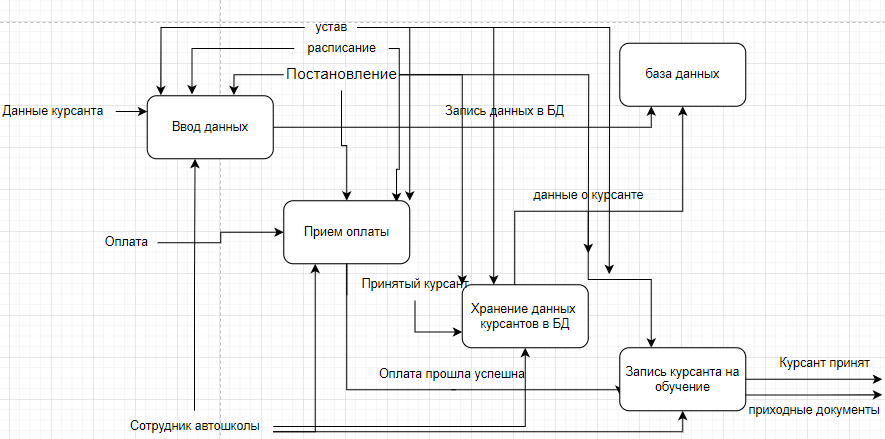


Рис.2.1. Диаграмма: прием новых курсантов



Рис.2.2 Диаграмма: процесс обучения

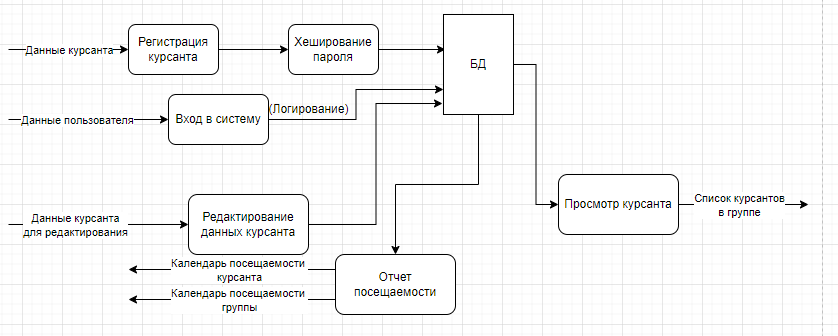


Рис. 2.3. Диаграмма: Регистрация курсанта

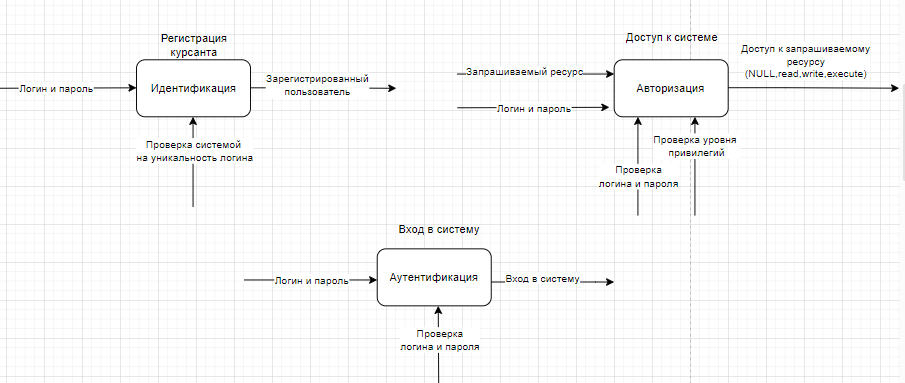


Рис. 2.4. Диаграмма: Идентификации, авторизации и аутентификации в системе

**Описание вариантов использования системы**

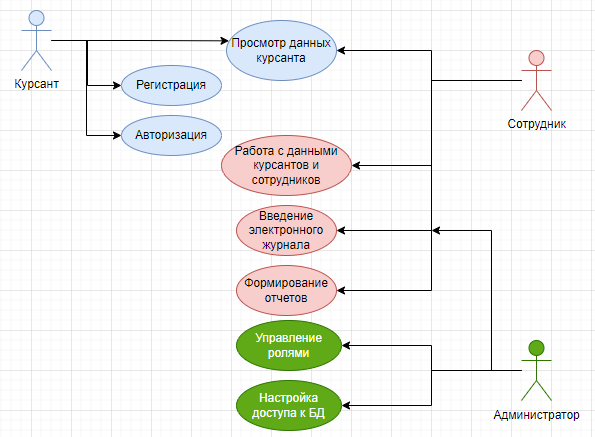


Рис. 2.5. Диаграмма Use Case

## **Организация структуры данных**

Схема «сущность-связь» (также ERD или ER-диаграмма) — это разновидность блок- схемы, где показано, как разные «сущности» (люди, объекты, концепции и так далее) связаны между собой внутри системы. ER-диаграммы чаще всего применяются для проектирования и отладки реляционных баз данных в сфере образования, исследования и разработки программного обеспечения и информационных систем. ER- диаграммы (или ER-модели) полагаются на стандартный набор символов, включая прямоугольники, ромбы, овалы и соединительные линии, для отображения сущностей, их атрибутов и связей. Эти диаграммы устроены по тому же принципу, что и грамматические структуры: сущности выполняют роль существительных, а связи — глаголов.

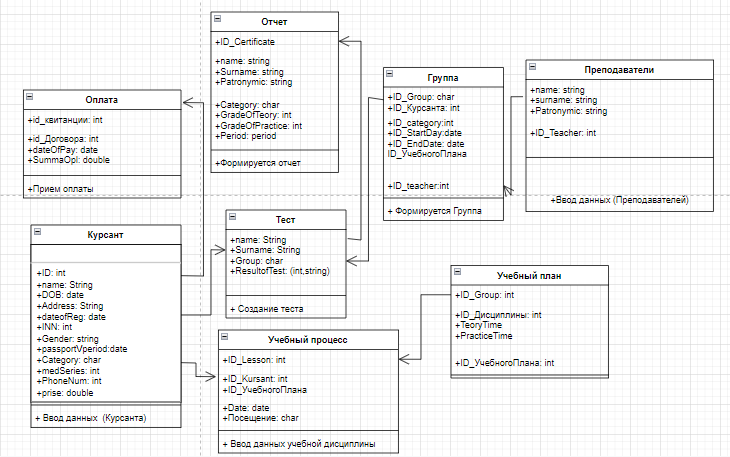


Рис. 2.5. ER - Диаграмма

## **Архитектура разрабатываемой системы**

Архитектура разрабатываемой системы основывается на монолитной.

Выражение «монолитная архитектура» сразу ассоциируется со словом «монолит». А монолитом еще с давних пор называют большой единый блок из камня или бетона. Монолит — это что-то большое и единое, имеющее общую и мощную структуру.

В программировании «монолитная архитектура» также подразумевает наличие общей и единой платформы, где сконцентрированы все компоненты одной программы. Сколько бы ни насчитывалось подобных компонентов, все они унифицированы и при этом управляются из одного места. В этом и определяется сила «монолитных» приложений.

Многие современные стартапы выбирают именно монолитную архитектуру приложения, потому что она комфортна при работе небольшими группами разработчиков. При ее использовании все компоненты программы взаимосвязываются и взаимозаменяются — это помогает развивать программу автономной и самодостаточной. Монолитная архитектура считается традиционной и проверенной при разработке приложений.

Монолитная архитектура хоть и «старая» по своему происхождению, но до сих пор актуальна и используется многими компаниями, так как идеально подходит для небольших стартапов и разработок, а также удовлетворяет следующим задачам:

* когда нужно быстро развернуть небольшое приложение;
* если в команде разработчиков небольшое количество людей (2-5), которые смогут работать совместно, а также смогут вместе поддерживать приложение в дальнейшем;
* когда создается непроверенный продукт и нужно его быстро создать, чтобы протестировать;
* когда просто нет опыта работы с микросервисами;
* если изначально известно, что приложение не будет разрастаться до колоссальных масштабов;
* если в приоритете разработки программного обеспечения находятся именно скорость его работы и производительность.

Поэтому перед использованием монолитной архитектуры нужно все тщательно взвесить. Иногда продумывают такой ход: запуск приложения осуществляют при использовании монолитной архитектуры, а чуть позже, если оно «зашло» пользователю, то его можно быстро раздробить на микросервисы для удобного масштабирования, тем самым меняя его архитектуру. Достоинства монолитной архитектуры являются простая разработка и запуск программы. Благодаря чему, вся разработка сконцентрирована в одном месте, легче дается интегрировать инструменты для облегченной разработки, а также при необходимости изменить элементы программы не нужно вносить изменения по отдельности в разных местах — все делается в одном месте.

Проблемы у монолитной архитектуры практически отсутствуют, так как большое количество приложений имеют зависимость от задач, которые совершаются между компонентами программы: логи, ограничения скорости, контрольные журналы и т. д. При монолитной архитектуре эти проблемы практически отсутствуют, так как все сконцентрировано в одном коде и все работает в одном приложении.

# **ГЛАВА 3. ПРОЕКТИРОВАНИЕ БЕЗОПАСНОСТИ**

## **3.1 Процессы организации**

Основным процессом организации является автоматизация документооборота, создание единой базы данных автошколы. Данный процесс состоит из следующих под-процессов:

* Регистрация
* Авторизация
* Выгрузка успеваемости курсанта
* Выгрузка расписания занятий
* Распределение нагрузок преподавателей
* Выгрузка отчетов

## **3.2 Инфраструктура клиент-серверного приложения**

Инфраструктура клиент-серверного приложения организации описана в виде функциональной схемы на рисунке 3.1

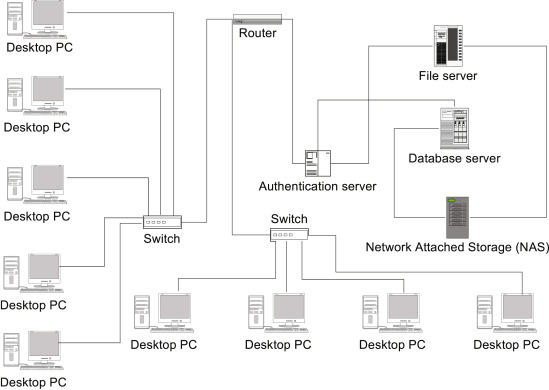


Рис.3.1. Архитектура сети приложения

## **3.3 Идентификация активов**

Согласно ГОСТ Р 53114-2008 «Защита информации. Обеспечение информационной безопасности в организации. Основные термины и определения»

Актив – все, что имеет ценность для организации в интересах достижения целей деятельности и находится в ее распоряжении.

Информационный актив – знания или данные, которые имеют значение для организации.

К активам организации могут относиться:

* Информационные активы, в том числе различные виды информации, циркулирующие в информационной системе (служебная, управляющая, аналитическая, деловая и т.д.) на всех этапах жизненного цикла (генерация, хранение, обработка, передача, уничтожение);
* Ресурсы (финансовые, людские, вычислительные, информационные, телекоммуникационные и прочие);
* Процессы (технологические, информационные и т.д.);

Выделены следующие активы, связанные с автошколой:

1. Информационный актив:
   1. Персональные данные
   2. Данные ограниченного доступа (отчеты, планы, группы, журналы и т.д.)
   3. Данные открытого доступа (учебные материалы и др.)
   4. Аутентификационные данные (логин/пароль)
2. Ресурсы:
   1. Программные:
      1. Операционные системы
      2. Программные обеспечения и файлы
   2. Аппаратные:
      1. Серверы
      2. Персональные компьютеры
      3. Сетевые устройства
   3. Помещения

## **3.4 Модель нарушителя**

Под моделью нарушителя понимаются предположения о возможностях нарушителя, которые он может использовать для разработки и проведения атак, а также об ограничениях на эти возможности. Модель нарушителя является важной частью информационной безопасности организации.

Все источники угроз безопасности информации можно разделить на три основные группы:

* Антропогенные источники угроз.
* Техногенные источники угроз.
* Стихийные источники угроз.

Антропогенными источниками угроз безопасности информации выступают субъекты, действия которых могут быть квалифицированы как умышленные или случайные преступления.

К техногенным источникам угроз относятся угрозы, которые вышли из-под контроля человека и существуют самостоятельно.

К стихийным источникам угроз относятся стихийные бедствия, которые невозможно предусмотреть и предотвратить.

Нарушитель типа «А»

К это й категории нарушителей относятся внутренние сотрудники организации, имеющие право работы с системой, а также имеющие к ней доступ. К ним можно отнести работников учреждения, которые в корыстных целях могут скомпрометировать персональные данные, хранящиеся в базе данных.

Нарушитель типа «Б»

Нарушителями типа «Б», являются компьютерными злоумышленниками (так же хакеры), которые пытаются удаленно получить доступ к системе с целью получения персональных данных от системы, используемых сотрудниками.

Нарушитель типа «В»

К нарушителям типа «В» могут относиться сотрудники организации, имеющие к нему как физический, так и логический доступ. К ним относятся системные администраторы, имеющие право и возможность настраивать параметры передачи данных с помощью системы, модифицировать параметры портов для считывания поступающей информации, имеющие возможность физического извлечения/замены элементов рабочего компьютера сотрудника.

## **3.5 Модель угроз**

Модель угроз ИБ – это описание существующих угроз ИБ, их актуальности, возможности реализации и последствий.

[ГОСТ Р 53114-2008](https://www.wikisec.ru/index.php?title=%D0%93%D0%9E%D0%A1%D0%A2_%D0%A0_53114-2008): модель угроз (безопасности информации): Физическое, математическое, описательное представление свойств или характеристик угроз безопасности информации.

Модели угроз информационной безопасности позволяют выявить существующие угрозы, разработать эффективные контрмеры, повысив тем самым уровень ИБ, и оптимизировать затраты на защиту (сфокусировав её на актуальных угрозах).

Составление модели угроз является необходимым при решении следующих задач:

* Выбор методов и способов защиты информации, хранящейся в серверах университета и передаваемой по сети;
* Формирование перечня мероприятий, направленных на предотвращение несанкционированного доступа (далее - НСД) к ресурсам внутренних серверов и (или) передачи данных, обрабатываемых в конечных узлах, лицам, не имеющим права доступа к такой информации;
* Предупреждение и предотвращение воздействия на физические компоненты, в результате которого может быть нарушено функционирование;
* Контроль над обеспечением уровня защищенности активов.
* Определение уровня защищенности веб-приложения, формирование перечня мероприятий, направленных на предотвращение несанкционированного доступа к ресурсам веб-приложения

Разработка Модели произведена на основании анализа исходных данных объекта исследования, нормативных и правовых документов органов исполнительной власти с учетом требований по безопасности, предъявляемым к информационным системам.

В модели должны учитываться все актуальные угрозы на всех стадиях их жизненного цикла. У различных информационных систем, а также объектов одной информационной системы может быть разный спектр угроз, определяемый особенностями конкретной информационной системы и её объектов и характером возможных действий источника угрозы.

Под уровнем исходной защищенности ИС понимается обобщенный показатель, зависящий от технических и эксплуатационных характеристик ИС. Исходная степень защищенности определяется экспертным путем, а именно:

* 0 – для высокой степени исходной защищенности;
* 5 – для средней степени исходной защищенности;
* 10 – для низкой степени исходной защищенности.

Под частотой (вероятностью) реализации угрозы понимается определяемый экспертным путем показателя, характеризующий, насколько вероятным является реализация конкретной угрозы безопасности ПД для данной ИС в складывающихся условиях обстановки. Вводятся четыре вербальных градации этого показателя:

* **маловероятно** – отсутствуют объективные предпосылки для осуществления угрозы (например, угроза хищения носителей информации лицами, не имеющими легального доступа в помещение, где последние хранятся);
* **низкая вероятность** – объективные предпосылки для реализации угрозы существуют, но принятые меры существенно затрудняют ее реализацию (например, использованы соответствующие средства защиты информации);
* **средняя вероятность** – объективные предпосылки для реализации угрозы существуют, но принятые меры обеспечения безопасности ПД недостаточны;
* **высокая вероятность** – объективные предпосылки для реализации угрозы существуют и меры по обеспечению безопасности ПД не приняты.

При составлении перечня актуальных угроз безопасности ПД каждой градации вероятности возникновения угрозы ставится в соответствие числовой коэффициент Y2, а именно:

* 0 – для маловероятной угрозы;
* 2 – для низкой вероятности угрозы;
* 5 – для средней вероятности угрозы;
* 10 – для высокой вероятности угрозы;

С учетом изложенного коэффициент реализуемости угрозы Y будет определяться соотношением:

Y = (Y1 +Y2)/ 20 (Ф-1)

Y1 – числовой коэффициент, определяющий исходную степень защищенности;  
Y2 – числовой коэффициент, определяющий вероятность возникновения угрозы;  
20 – нормирующий коэффициент.

Коэффициент Y1 определяется, исходя из набора технических и эксплуатационных характеристик, общих для всех ИСПД. Коэффициент Y2 определяется на основе экспертной (субъективной) оценки вероятности возникновения угрозы.

По значению коэффициента реализуемости угрозы Y формируется вербальная интерпретация реализуемости угрозы следующим образом:

* если 0 <= Y <= 0,3, то возможность реализации угрозы признается низкой;
* если 0,3 <Y <= 0,6, то возможность реализации угрозы признается средней;
* если 0,6 <Y <= 0,8, то возможность реализации угрозы признается высокой;
* если Y> 0,8, то возможность реализации угрозы признается очень высокой;

Затем осуществляется выбор из общего (предварительного) перечня угроз безопасности тех, которые относятся к актуальным для данной ИС, в соответствии с правилами, приведенными в таблице 3.1

Таблица 3.1. Правила отнесения угрозы безопасности ПД к актуальной

|  |  |  |  |
| --- | --- | --- | --- |
| Возможность  реализации  угрозы | Показатель опасности угрозы | | |
| Низкая | Средняя | Высокая |
| Низкая | неактуальная | неактуальная | неактуальная |
| Средняя | неактуальная | актуальная | актуальная |
| Высокая | Актуальная | актуальная | актуальная |
| Очень высокая | Актуальная | актуальная | актуальная |

Таблица 3.2. Определение актуальных угроз безопасности

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Угроза** | **Уровень защищенности (Y1)** | **Вероятность реализации угрозы (Y2)** | **Коэффициент реализуемости угрозы (Y)** | **Актуальность** |
| Повреждения программных и технических средств | 0 | 2 | 0,1 | неактуальная |
| Получение уровень доступа админа | 0 | 2 | 0,1 | неактуальная |
| Сбои и отказы в обслуживании | 5 | 10 | 0,75 | актуальная |
| SQL-инъекции | 10 | 10 | 1 | актуальная |
| Перехват аутентификационных данных | 10 | 5 | 0,75 | актуальная |
| Обход контроля доступа | 5 | 5 | 0,5 | актуальная |
| Повышение  привилегий | 5 | 10 | 0,75 | актуальная |

**3.6 Способы защиты от вероятных атак**

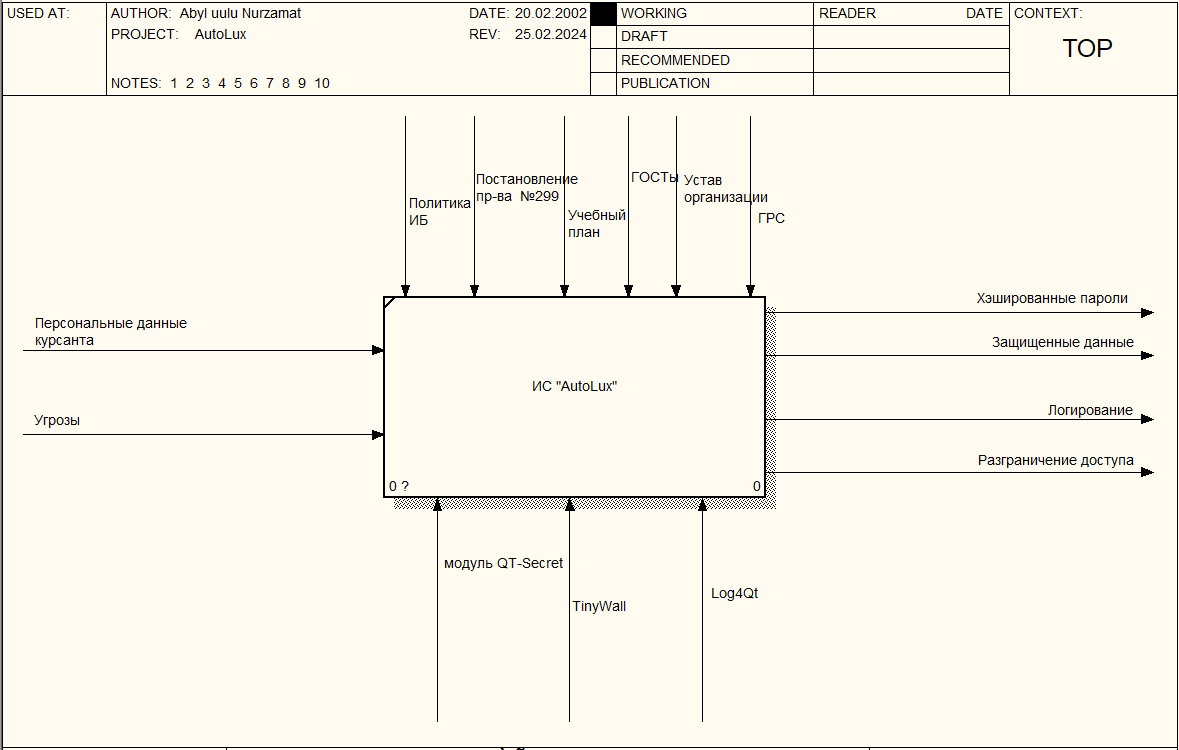


Рис. 3.1 Диаграмма IDEF0

Входные данные:

- Персональные данные

- Угрозы

Управление:

- Политика ИБ

- Модель угроз

- Модель нарушителя

- ГОСТ ИСО/МЭК 27001

- ГОСТ Р 53114-2008

Механизмы:

- Log4Qt

- TinyWall

- модуль QT-Secret

Выходные данные:

- Хэшированные пароли

- Защищенные данные

- Разграничение доступа

- Логирование

### **3.6.1 Log4Qt - Логирование**

Log4Qt - это порт C++ пакета Log4j Apache Software Foundation с использованием Qt Framework. Он предназначен для использования в проектах с открытым исходным кодом и коммерческих проектах Qt.

***Дополнительные возможности:***

* SimpleTimeLayout («дд.ММ.гггг чч:мм Сообщение регистратора уровня [Thread]»)
* ColorConsoleAppender (рендеринг цветного сообщения с помощью escape-последовательности и вывод его на консоль)
* SignalAppender (выдает сигнал при возникновении события журнала)
* DatabaseAppender (добавление события журнала в таблицу sql)
* DatabaseLayout (поместите событие журнала в столбцы таблицы sql)
* Приложение Telnet (добавление событий журнала к клиентам Telnet)
* LogStream (добавление сообщений журнала в стиле qDebug())
* MainThreadAppender (прокси-приложение для отправки сообщений журнала через цикл событий)
* XMLLayout для поддержки бензопилы Apache
* DailyFileAppender, который генерирует файл журнала за каждый день (добавьте текущую дату в формате имени файла)
* Двоичный регистратор
* Приложение консоли отладки Windows

Для подключения логгера Log4QT нам нужно сперва добавить заголовочный файл log4qt.pri в наш проект Рис.

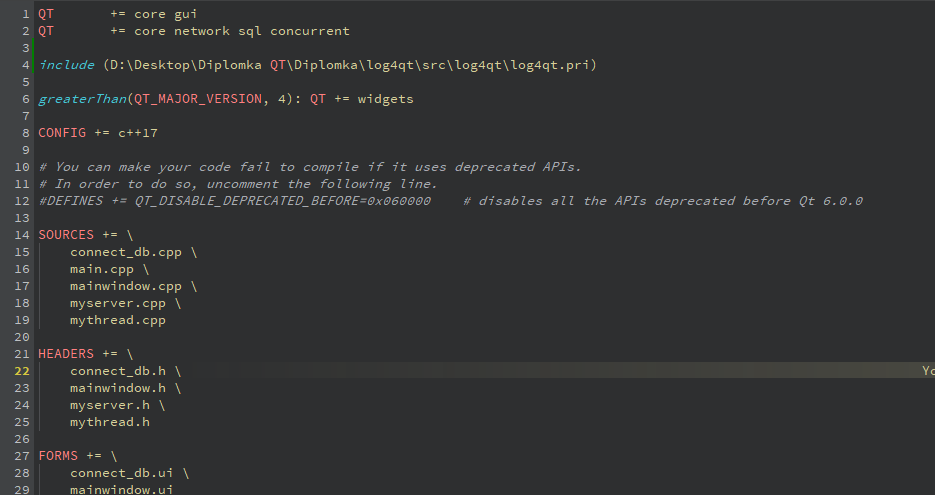


Рис. Подключение log4qt

Далее подключаем следующие классы:

1. include "log4qt/consoleappender.h"
2. include "log4qt/logger.h"
3. include "log4qt/ttcclayout.h"

Настраиваем регистратор для генерации выходных данных Рис.



Рис. Настройка корневого регистратора для примера

### **3.6.2 TinyWall - FireWall**

**TinyWall** - инструмент для повышения уровня сетевой защиты операционной системы.

После установки программа автоматически блокирует все исходящие соединения, кроме нескольких доверенных программ, например, (IE, Firefox и так далее). Если вы хотите выйти в Интернет с помощью приложения, которое не входит в доверенный перечень программ, то связь по-умолчанию блокируется без всплывающих окон предупреждений.

Настройка TinyWall на такое поведение - блокировку по-умолчанию без предупреждений, является преднамеренным, так как автор программы считает, что когда появляется окно предупреждения Разрешить / Запретить, пользователь чаще нажимает Разрешить. Чтобы разрешить соединение в TinyWall необходимо нажать правой кнопкой мыши на значке программы в трее панели задач, выбрать "В белый список..." и затем выбрать нужную программу из списка. После этого сразу будет доступно исходящее соединение, а утилита TinyWall запомнит ваш выбор.

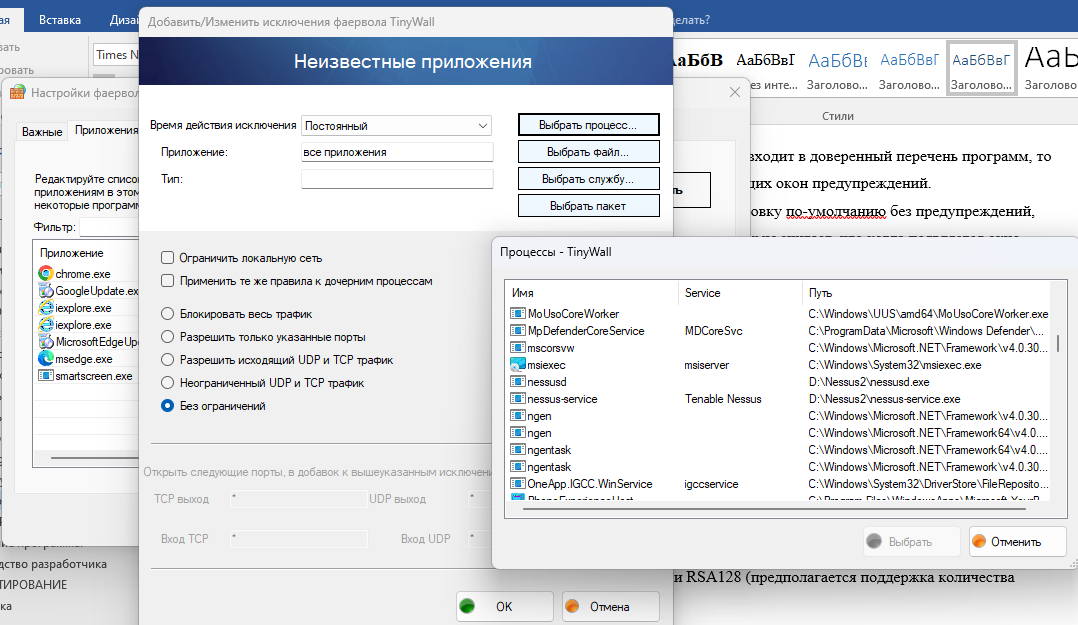


Рис. Настройка firewall

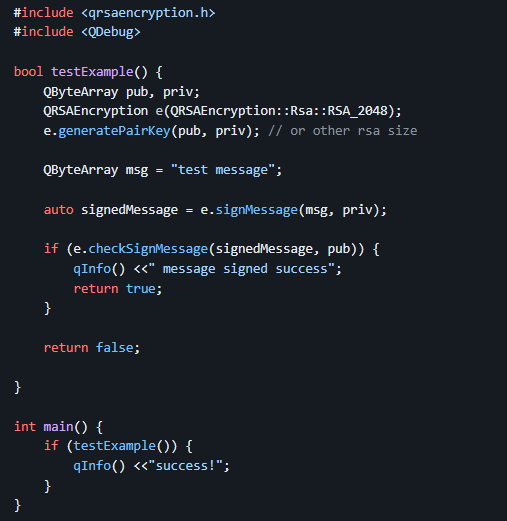
### **3.6.3 Модуль QT-Secret**

[Qt-Secret](https://github.com/QuasarApp/Qt-Secret)— это библиотека, созданная [группой QuasarApp](https://github.com/QuasarApp)на Qt/qmake, цель — для обеспечения основных возможностей шифрования, которых нет в родном Qt. А именно: алгоритмы RSA и AES.

Ключевая особенность:

* Генерация пар ключей RSA64 и RSA128 (предполагается поддержка количества номеров до RSA2048)
* Шифрование и дешифрование RSA.
* Подпись и аутентификация сообщения.
* Генерация ключей AES (AES64, AES128, AES256)
* Шифрование и дешифрование AES

****

****

**ГЛАВА 4. РАЗРАБОТКА ДОКУМЕНТАЦИИ**

## **4.1 Описание программы**

Разрабатываемое клиент-серверное приложение предназначено для автоматизации процессов автошколы.

В системе есть три вида пользователей – администратор, преподаватель, курсант.

Администратор имеет полный доступ к системе, кроме выставления четвертных оценок. Может создать учетные данные к завучам.

Преподаватель имеет привилегированный доступ к системе, может: создать группы курсантов, загрузить расписание занятий, выгрузить отчеты и так далее.

Курсант имеет право только на просмотр своих данных а также занятий.

## **4.2 Руководство разработчика**

**4.2.1 Назначение и условия применения**

Данная система предназначена для управления проектами по разработке программного обеспечения. Она представляет с собой клиент-серверное приложение

Требования к составу и параметрам технических средств:

* операционная система: Windows, MacOS, Linux;
* процессор: Тактовая частота от 2300 МГц и выше;
* ОЗУ: 8 и выше;
* свободное место на диске: не менее 256 Гб.

**4.2.2 Средства реализации**

В качестве средств реализации веб-приложения были использованы следующие технологии:

* язык программирования Java от 8 версии;
* фреймворк для создания веб-приложения Spring boot;
* сборщик приложения Gradle;
* система управления базами данных: MySQL;

**4.2.3 Входные и выходные данные**

Система получает на вход следующие данные:

* данные регистрации;
* запросы на регистрацию;
* данные авторизации;
* запросы на редактирование персональных данных;
* данные сотрудников;
* данные учеников;
* данные отчетов;
* редактированные персональные данные.

А в качестве выходных:

* список классов;
* список учеников;
* список расписаний;
* список пользователей;
* список достижений;
* список мероприятий;
* список неподтвержденных пользователей на регистрацию;
* список неподтвержденных пользователей на изменение данных;
* информационные сообщения об ошибках.

**ГЛАВА 5. ТЕСТИРОВАНИЕ**

Тестирование информационной системы – это процесс исследования, испытания программного обеспечения для проверки на соответствие между реальным и ожидаемым поведением на конечном наборе тестов.

И в процессе тестирования на отказоустойчивость системы необходимо проделать действия, которые помогут проверить информационную систему в различных сценариях использования, и как следствие установить, степень работоспособности приложения.

**5.1 DoS атака**

ModSecurity- это бесплатный модуль для Apache с открытым кодом, организующий функции firewall. Он позволяет организовать защиту от большого количества атак против web-сайтов, используя мониторинг и анализ всех запросов в реальном времени

Для этого модуля есть большое число [правил](https://www.owasp.org/index.php/Category:OWASP_ModSecurity_Core_Rule_Set_Project), написанных сообществом для предотвращения популярных атак

Мы же будем использовать этот модуль для блокирования DDoS атак

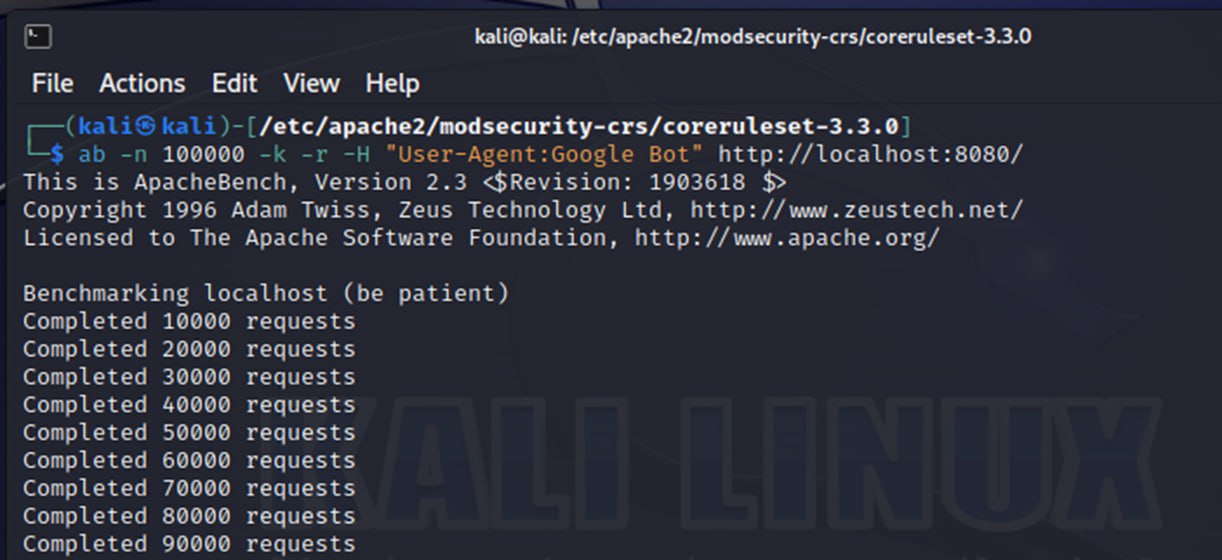
****

Рис.5.1. Отправка запросов. Реализация Dos атаки

На Рисунке 5.2 можно увидеть, что ModSecurity заблокировал запросы. При настройке ModSecurity нужно подключить отдельный модуль, направленный на блокировку таких атак

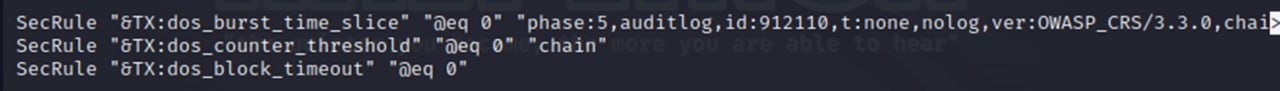
****

Рис. 5.2. Блокировка DoS атаки

**5.2 SQL инъекция**

SQL-инъекция (SQL Injection) — это вид атаки на веб-приложения, при которой злоумышленник внедряет вредоносный SQL-код в строку запроса, предназначенную для выполнения SQL-запросов к базе данных. Целью атакующего является получение несанкционированного доступа к данным или выполнение опасных операций в базе данных.

SQL-инъекции возникают, когда веб-приложения недостаточно проверяют и экранируют пользовательский ввод, прежде чем использовать его в SQL-запросах. Злоумышленник может внедрить вредоносный SQL-код, используя различные методы, включая ввод веб-форм, параметры URL или HTTP-заголовки.

Примеры возможных последствий SQL-инъекций включают:

* Получение доступа к конфиденциальным данным: Злоумышленник может использовать SQL-инъекцию для извлечения, изменения или удаления данных из базы данных. Это может включать получение личных данных пользователей, паролей, финансовой информации и т. д.
* Выполнение несанкционированных операций: Атакующий может использовать SQL-инъекцию для выполнения опасных операций в базе данных, таких как удаление или изменение данных, создание новых учетных записей или даже уничтожение базы данных.
* Обход авторизации и аутентификации: Злоумышленник может использовать SQL-инъекцию для обхода механизмов аутентификации и авторизации, получая доступ к защищенным разделам приложения или выполняя действия от имени других пользователей.

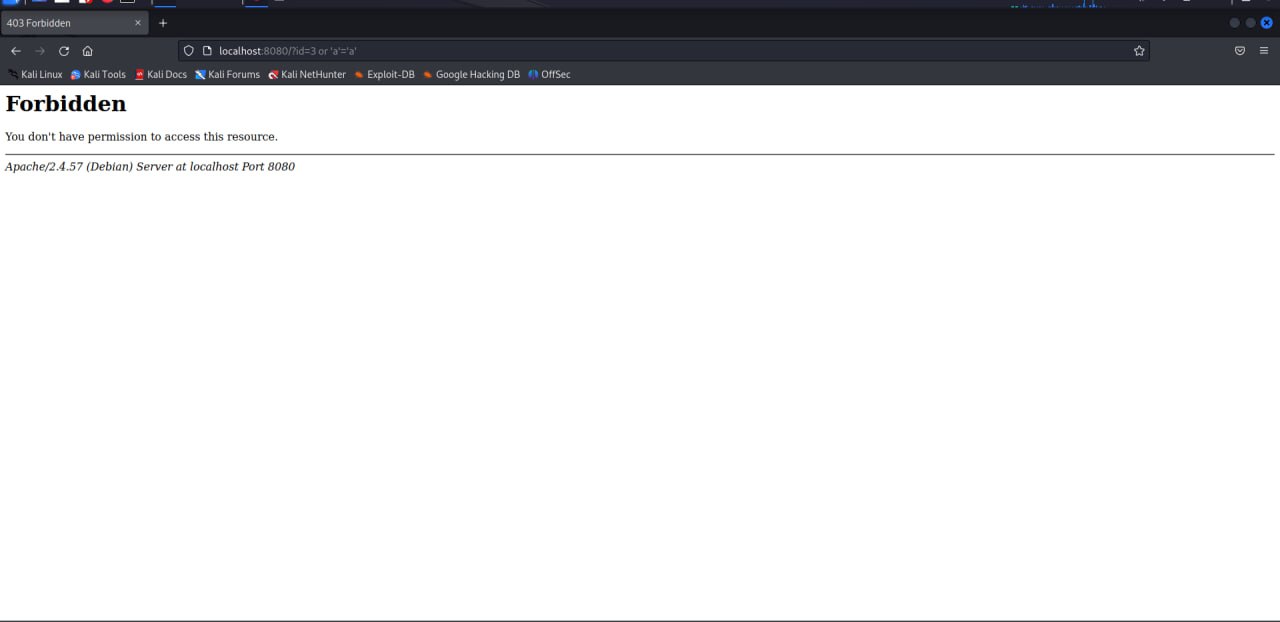


Рис.5.3. Реализация SQL инъекции

На рисунке 5.4. можно увидеть, как, ModSecurity выявил и предотвратил нашу атаку. Также отдельно устанавливается модуль OWASP-TOP10.

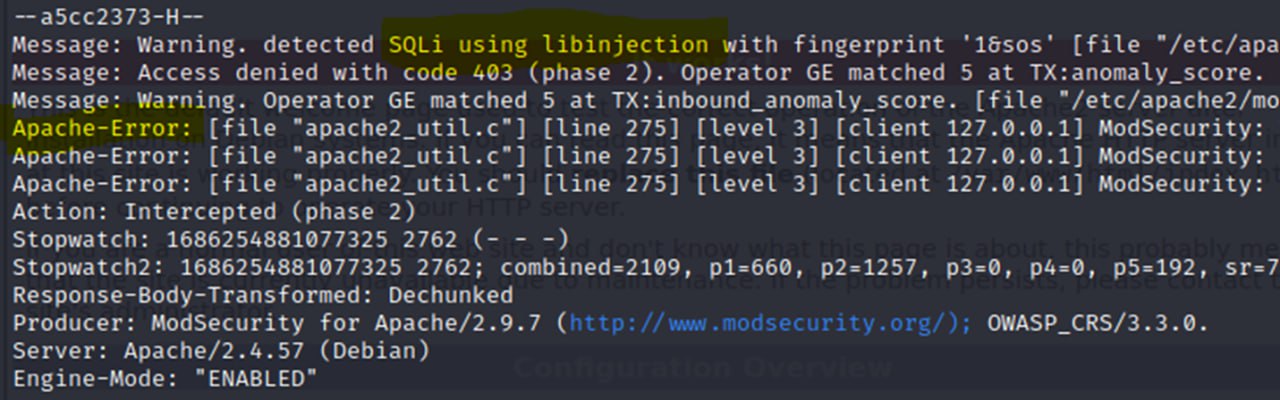


Рис.5.4. Отчет ModSecurity

**ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ**

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ**

ЗАКЛЮЧЕНИЕ

# **ПРИЛОЖЕНИЯ**

ПРИЛОЖЕНИЕ А

**Идентификация активов**

Таблица 1. П-А. Идентификация активов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Бизнес-процесс** | **Активы** | **Среда обработки** | **ПСИБ** | **Уязвимости** |
| Регистрация | Персональные данные (ИА-1) | QT V6 (ПА) | К,Ц,Д | CVE-2021-1636 |
| MS SQL Server 2017 CU31 (ПА) | CVE-2022-23298  CVE-2019 11137 |
| Авторизация | Идентификационные данные (ИА-2) | QT V6 (ПА) | К,Ц,Д | CVE-2022-29143  CVE-2017-8516 |
| MS SQL Server 2019 CU31 (ПА) | CVE-2022-24454  CVE-2019-11137 |
| Редактирование аккаунтов пользователей | База данных (ИА -3) | Microsoft SQL Server 2019 (RTM) 15.0.2000.5 Enterprise Edition | Д,Ц,К | CVE-2022-29143  CVE-2017-8516 |
|  | MS Office 2019 16.0 | Д,Ц,К | CVE-2022-24454 |
| 15.0.2000.5 Enterprise Edition (ПA)  MS Office 2019 16.0 (ПA) | QT V6 (ПА) | Д,Ц,К | CVE-2019-11137 |
| Обработка информации | База данных (ИА-3) | Microsoft SQL Server 2019 (RTM) 15.0.2000.5 Enterprise Edition | Ц,Д,К | CVE-2022-34006 |

Продолжение таблицы 1. П-А. Идентификация активов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Нагрузка  преподавателей | База данных (ИА-3) | Microsoft SQL Server 2019 (RTM) 15.0.2000.5 Enterprise Edition | Ц,Д,К | CVE-2022-34006 |
| Microsoft SQL Server 2019 (RTM) | MS Office 2019 16.0 | Ц,Д,К | CVE-2022-3140 |
| Windows 10 Enterprise 21H2 (ПA) | QT V6 (ПА) | Ц,Д,К | CVE-2022-23298 |
| Отчет | База данных (ИА-3) | Microsoft SQL Server 2019 (RTM) 15.0.2000.5 Enterprise Edition | Ц,Д,К | CVE-2022-34006 |
| Microsoft SQL Server 2019 (RTM) | MS Office 2019 16.0 | Ц,Д,К | CVE-2022-3140 |
| Windows 10 Enterprise 21H2 (ПA) | QT V6 (ПА) | Ц,Д,К | CVE-2022-23298 |

ПРИЛОЖЕНИЕ Б

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Таблица 2.П-Б. Модель нарушителя | | | | | | |
| **Тип нарушителя** | **Вид нарушителя** | **Описание уровня доступа** | **Мотивация нарушителя** | **Квалификация, знания и ресурсы нарушителя** | **Реализуемые угрозы** | **Способы реализации угроз** |
| Антропогенный,  Тип А  (Сотрудник, не имеющий доступа) | Внутренний нарушитель | Физический  доступ к системе | Корыстный  умысел,  месть,  любопытство | Владение  информацией об  аппаратном и  программном  оснащении  компьютера | НСД к  персональному  компьютеру | Копирование/перемещение информации,  представляющей  ценность для  организации на  внешний носитель |
| Внутренний нарушитель | Заражение ОС  троянской программой.  Перебор пароля  пользователя по  словарю при  удаленном  подключении к  компьютеру |
| Продолжение таблицы 2. П-Б. Модель нарушителя | | | | | | |
| Антропогенный,  Тип Б  (Компьютерный  злоумышленник) | Внешний | Доступ по  протоколу IP к  интернет-шлюзу  со стороны  интернет-сети | Корыстный  умысел, месть,  любопытство, вандализм | Знание стека  сетевых  протоколов  TCP/IP, знание  IP-адреса  интернет-шлюза | Сбои, отказы,  разрушения/повреждения  программных и  технических  средств | DoS-атака |
| Знание языка  SQL, протоколов  HTTP/HTTPS | SQL-инъекция | Внедрение  вредоносного sql кода в  формы или строку  запроса |
| Владение  программными  средствами  перехвата пакетов в IP  сетях, знание IP-  адреса интернет-  шлюза организации в  сети «интернет» | Перехват  аутентификационных данных | Компрометация  интернет-шлюза  (подбор  идентификатора  последовательности и  номера порта-  отправителя) |
|  |  |  |  |  |  |  |
| Продолжение таблицы 2. П-Б. Модель нарушителя | | | | | | |
|  | Внешний | Удаленный доступ к серверу  системы,  персональному  компьютеру  сотрудника | Корыстный  умысел, месть,  любопытство, вандализм | Наличие  электронного  почтового ящика.  Навыки  использования  электронной  почты. Знание стека протоколов TCP/IP | Сбой/отказ;  Компрометация данных, передаваемых посредством интернета | Перебор пароля  пользователя по  словарю при  удаленном подключении к компьютеру с  вредоносным  вложением; |
| Антропогенный,  Тип В  (Сотрудник организации, имеющий право работы с системой, а также имеющие к нему  физический доступ) | Внутренний нарушитель | Физический и  логический  доступ к серверу  системы,  персональному  компьютеру  сотрудника | Корыстный  умысел, месть,  любопытство, халатное отношение | Владение  информацией об  аппаратном и  программном  оснащении  компьютера | НСД к  персональному  компьютеру | Заражение ОС  троянской программой. |
| Внутренний нарушитель | Владение  информацией об  аппаратном и  программном  оснащении сервера | Удаление/модификация данных, хранящихся на  сервере | Запуск исполняемого  файла с цифровой  подписью доверенного  приложения |
|  |  |  |  |  |  |  |
| Продолжение таблицы 2. П-Б. Модель нарушителя | | | | | | |
| Антропогенный,  Тип В  (Уволенные сотрудники) | Внутренний нарушитель | Доступ по  протоколу IP к  интернет-шлюзу  со стороны  интернет-сети | Корыстный умысел, месть, вандализм | Владение  информацией об  аппаратном и  программном  оснащении  компьютера | Компрометация  данных,  передаваемых  посредством  интернета | Копирование/перемещение информации,  представляющей  ценность для  организации на  внешний носитель |

ПРИЛОЖЕНИЕ В

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Таблица 3.П-В. Модель угроз | | | | | | | |
| **Угроза ИБ** | **Источник угрозы ИБ** | **Актив** | | | | **Метод реализации угрозы**  **ИБ на среду обработки ИА** | **Последствия реализации угрозы ИБ** |
| **Информационный актив** | **Значимые свойства ИБ в порядке приоритета** | **Среда обработки ИА** | **Уязвимость среды обработки ИА** |
| Сбои и отказы в  обслуживании | Антропогенный тип Б | Информация открытого доступа | Доступность  Целостность  Конфиденциальность | QT framework v6 | BDU:2021-02185, BDU:2020-02766 | DDOS-атака. Посылание  большого количества  вредоносных запросов на  сервер. | Остановка  работы сервера и выдачи  ошибки отказ в  обслуживании |
| SQL-инъекции | Персональные данные и данные ограниченного доступа | Целостность  Конфиденциальность  Доступность | SQL Server 2019 | BDU:2019-01225 | SQL-инъекция. Внедрение  вредоносного SQL-кода в  запросы к информационной системе | Нарушение целостности и  конфиденциальности |

Продолжение таблицы 3. П-В. Модель угроз

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Повышение  привилегий | Антропогенный  Тип В | Данные ограниченного доступа | Конфиденциальность  Целостность  Доступность | QT framework v6 | BDU:2021-02185, BDU:2020-02766 | Повышение  привилегий —эксплуатация уязвимостей в  операционной системе или  прикладном ПО | Нарушение  конфиденциальности и  целостности |
| Кража  аутентификационных данных | Антропогенный  Тип В | Персональные данные | Конфиденциальность  Целостность Доступность | QT framework v6 | BDU:2021-02185, BDU:2020-02766 | Атака человек по середине | Нарушение  конфиденциальности |
| Обход контроля доступа | Антропогенный  Тип Б | Данные ограниченного доступа,  Персональные данные | Доступность  Конфиденциальность  Целостность | QT framework v6 | BDU:2023-03689 | SQL-атака, подмена, URL-адреса | Нарушение конфиденциальности и доступности |
|  |  |  |  |  |  |  |  |