Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Сибирский государственный автомобильно-дорожный университет (СибАДИ)»

Институт Информационные системы, экономика и управление

Кафедра Информационная безопасность

Специальность Информационная безопасность автоматизированных систем

Специалитет Информационная безопасность автоматизированных систем на транспорте

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**к курсовой работе**

по дисциплине: «Безопасность систем баз данных»

Тема работы «Защита данных в информационной системе «Автосервис» (на примере использования шифрования данных и ролевой политики в СУБД PostgreSQL и защиты клиентского приложения на ЯП Python)»

Выполнил:

студент группы БИ-20Э1 Д.А Долматов

Курсовая работа сдана на проверку:

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2023 г.

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д.А. Долматов

(подпись студента)

Курсовая работа допущена к защите:

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2023 г.

Руководитель работы

профессор кафедры, д.п.н. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ З.В. Семенова

(подпись преподавателя)

Курсовая работа защищена с оценкой:

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2023 г.

Руководитель работы

профессор кафедры, д.п.н. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ З.В. Семенова

Омск 2023

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Сибирский государственный автомобильно-дорожный университет (СибАДИ)»

Кафедра «Информационная безопасность»

УТВЕРЖДАЮ

И.О. Зав. кафедрой ИБ, к.т.н.,

\_\_\_\_\_\_\_\_\_\_\_\_\_ Д. А. Панков

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2023 г.

**ЗАДАНИЕ**

**к курсовой работе**

студенту группы БИ-20Э1 Долматову

1. Тема курсовой работы: «Защита данных в информационной системе «Автосервис» (на примере использования шифрования данных и ролевой политики в СУБД PostgreSQL и защиты клиентского приложения на ЯП Python)»
2. Исходные данные к курсовой работе:

* автоматизируемые процессы в информационной системе
* механизмы шифрования;
* количество связанных таблиц не менее 8 таблиц;
* количество запросов на обработку данных не менее 8;
* используемые средства - серверная часть PosqtgreSQL, клиент ЯП Python
* документация ЯП Python;
* документация СУБД PostgreSQL;

1. Содержание пояснительной записки:

* титульный лист;
* задание к курсовой работе;
* содержание;
* введение;
* теоретические аспекты проектирования ИС «Автосервис» и защиты данных;
* реализация ИС «Автосервис» и защита данных в ней;
* документация на ИС «Автосервис»;
* заключение;
* список использованных источников;
* приложения.

1. Перечень демонстрационного материала для сопровождения доклада при защите курсовой работы:

* презентация Microsoft PowerPoint.

1. Срок сдачи работы: «\_\_» \_декабря\_ 2023 г.

Задание выдано « » сентября 2023 г.

Руководитель работы: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ З.В. Семенова

(подпись преподавателя)

Задание к исполнению принял « » сентября 2023 г.

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д.А. Долматов

(подпись студента)

# СОКРАЩЕНИЯ И ОПРЕДЕЛЕНИЯ

ЯП – Язык программирования.

ПО – Программное обеспечение.

БД – База данных.

ИС – Информационная система.

Оглавление

[СОКРАЩЕНИЯ И ОПРЕДЕЛЕНИЯ 4](#_Toc157413166)

[ВВЕДЕНИЕ 6](#_Toc157413167)

[**1.** **ТЕОРЕТИЧЕСКИЕ АСПЕКТЫ РАЗРАБОТКИ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ «» И ЗАЩИТЫ ДАННЫХ В НЕЙ** 7](#_Toc157413168)

[**1.1.** **Описание основных бизнес функций и запросов в АС «Автосервис»** 7](#_Toc157413169)

[2. Практические аспекты разработки ас/ис «АВТОСЕРВИС» и ее защиты 18](#_Toc157413170)

[**2.1.** **Реализация серверной части** 18](#_Toc157413171)

# ВВЕДЕНИЕ

Целостность, доступность, конфиденциальность являются столпами безопасности информации. Эти три фактора также являются неотъемлемым требованием к современным базам данных, хранящих в себе терабайты ценной информации. Именно эти данные привлекают злоумышленников, для использования их в злых умыслах.

За первые четыре месяца 2023 года произошло 75 утечек из российских коммерческих компаний и госорганизаций. Это в 1,5 раза больше по сравнению с аналогичным периодом прошлого года, когда зафиксировали 49 утечек, пишут «Ведомости» со ссылкой на гендиректора F.A.C.C.T. (бывшая Group-IB в России и СНГ) Валерия Баулина.

Тренд подтвердили основатель сервиса разведки утечек данных и мониторинга даркнета DLBI Ашот Оганесян и аналитик Kaspersky Digital Footprint Intelligence Игорь Фиц. По их оценке, количество утечек в первом квартале возросло в 2 и 2,5 раза соответственно.

Оказавшиеся в сети данные принадлежали в основном крупнейшим финансовым и страховым компаниям, госструктурам, а также компаниям в сфере информационной безопасности, перечислил Баулин. По словам Оганесяна, в числе крупнейших похищений информации в 2023 г.оду оказались двойная утечка данных бонусной программы «Сберспасибо» (суммарно 52,5 млн записей), сети магазинов «Спортмастер» (46 млн записей) и интернет-аптеки zdravcity.ru (8,9 млн записей).

В апреле, по данным DLBI, произошли утечки данных из сервиса по продаже билетов kassy.ru (4,5 млн уникальных e-mail и столько же телефонов), интернет-магазина zoloto585.ru (9,9 млн записей) и страховой компании «Согаз» (8,3 млн записей).

Несмотря на то что утекать данные стали чаще, общее количество строк в утекших базах данных за четыре месяца 2023 года снизилось и составило примерно 65 млн, заметил Баулин. В тот же период 2022-го их число составляло 100 млн, добавил он.[1]



Рисунок -

Все эти недавние атаки показали, что безопасность систем баз данных требуется обеспечивать в полной мере и с использованием возможных средств, одними из таких средств являются использование ролевой политики и шифрование данных, которые позволят оградить охраняемый субъект.

Помимо угрозы хакерской атаки и утечки важных конфиденциальных данных, существуют определенные требования к безопасности информационной системы:

* Приказ ФСТЭК России от 18 февраля 2013 г. N21 «Об утверждении состава и содержания организационных и технических мер по обеспечению безопасности персональных данных при их обработке в информационных системах персональных данных» [2];
* Федеральный закон "О персональных данных" от 27.07.2006 N 152-ФЗ (последняя редакция) [3];

**Цель курсовой работы:** Обеспечение безопасности информационной системы «Автосервис» с использованием механизмов защиты данных СУБД PostgreSQL и клиентской программы.

**Задачи курсовой работы:**

1. Описать предметную область и определить основные бизнес-процессы.
2. Cформулировать запросы, необходимые для реализации в информационной системе, удовлетворяющие потребности бизнес-процессов.
3. Исследовать возможности обеспечения безопасности средствами СУБД PostgreSQL и ЯП Python.
4. Реализовать АС с использованием практических аспектов безопасности. Изучить способы обеспечения безопасности данных СУБД PostgreSQL
5. Реализовать основные механизмы защиты АС «Автосервис».
6. Подготовить руководство по использованию ИС.
7. **ТЕОРЕТИЧЕСКИЕ АСПЕКТЫ РАЗРАБОТКИ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ «Автосервис» И ЗАЩИТЫ ДАННЫХ В НЕЙ**
   1. **Описание основных бизнес-процессов и запросов в АС «Автосервис»**

Автосервис заботится о техническом обслуживании и ремонте вашего автомобиля. Под техническим обслуживанием понимается комплекс мероприятий, направленных на поддержание исправного состояния транспортного средства в соответствии с рекомендациями производителя. А ремонт включает в себя операции по восстановлению исправности автомобиля и его компонентов.

Опытные специалисты с опытом более трех лет гарантируют качественный ремонт. Они постоянно повышают свою квалификацию, участвуя в обучающих курсах.

Автосервис предоставляет услуги ремонта как для частных автовладельцев, так и для корпоративных клиентов. Принципы работы включают в себя многоуровневый контроль качества, соблюдение сроков ремонта и индивидуальный подход к каждому автомобилю.

Чтобы оформить заказ на на услуги, требуется предоставить следующую информацию: ФИО, адрес, телефон, а также данные об автомобиле, такие как VIN код, марка, модель, год выпуска и номер СТС.

Мы специализируемся на ремонте японских автомобилей и работаем с владельцами автомобилей.

В нашем автосервисе существует несколько ключевых бизнес-процессов, включая учет обращений клиентов, оформление заявок и предоставление ремонтных услуг. Процесс учета обращений клиентов начинается с заявки от клиента, после чего администратор получает согласие, собирает и проверяет

данные клиента, а затем добавляет информацию в базу данных.

Рисунок – процесс «Учет обращений клиентов»

Процесс «Оформление заявок на предоставление услуг», представлен на рисунке 3.2. Стартовым событием является подтверждение факта владения или права управления автомобилем, опираясь на один из фактов администратор собирает информацию об автомобиле, которая нужна для ремонта и добавляет авто в БД.

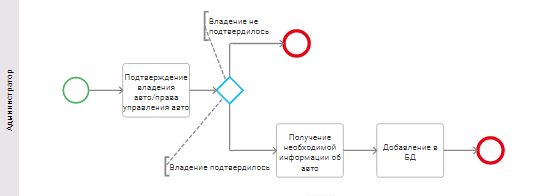


Рисунок - процесс «Оформление заявок на предоставление услуг»

Процесс «Ремонтные услуги» (рисунок 3.3). Стартовым событием является информация о цели визита, т.е для получения какого типа услуг клиент прибыл в сервисный центр. Если целью визита стало прохождение технического обслуживания, то мастер-приемщик определяет регламент технического обслуживания автомобиля, далее автомобиль проходи плановое технического обслуживания, затем администратор принимает оплату. Если целью визита стал ремонт автомобиля, то сначала мастер-приемщик проводит осмотр и диагностику авто, заполняет диагностическую карту, фиксируя в ней неисправности, составляет заявку на необходимые для ремонта запчасти, проверяет склад на наличие нужных деталей, если их нет, то происходит составление заявки на заказ у поставщика, затем мастер передает заявку администратору. Администратор согласовывает заявку с клиентом, опираясь на его ответ, он либо оформляет эту заявку, либо заканчивает процесс и принимает оплату за выполненную услугу. После получения необходимых деталей в работу вступают автомеханики, которые осуществляют ремонт автомобиля.

Процесс «Ремонтные услуги» (рисунок 3.3). Стартовым событием является информация о цели визита, т.е для получения какого типа услуг клиент прибыл в сервисный центр. Если целью визита стало прохождение технического обслуживания, то мастер-приемщик определяет регламент технического обслуживания автомобиля, далее автомобиль проходи плановое технического обслуживания, затем администратор принимает оплату. Если целью визита стал ремонт автомобиля, то сначала мастер-приемщик проводит осмотр и диагностику авто, заполняет диагностическую карту, фиксируя в ней неисправности, составляет заявку на необходимые для ремонта запчасти, проверяет склад на наличие нужных деталей, если их нет, то происходит составление заявки на заказ у поставщика, затем мастер передает заявку администратору. Администратор согласовывает заявку с клиентом, опираясь на его ответ, он либо оформляет эту заявку, либо заканчивает процесс и принимает оплату за выполненную услугу. После получения необходимых деталей в работу вступают автомеханики, которые осуществляют ремонт автомобиля .

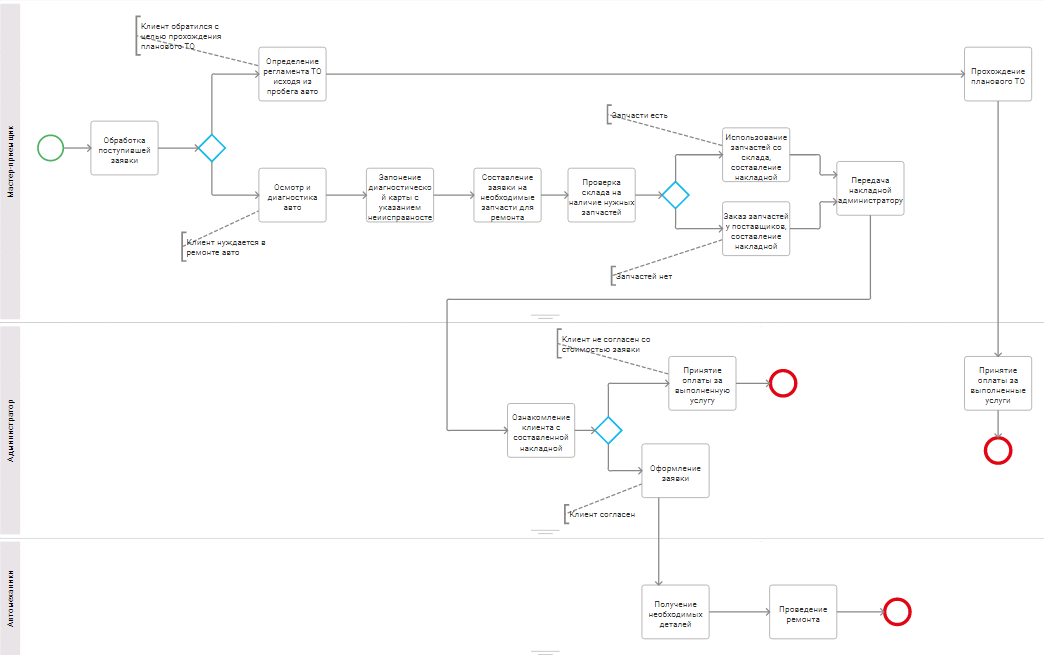


Рисунок 3.3 - процесс «Ремонтные услуги

Из схемы следует, что в системе должны быть запросы:

* + 1. Получение автомобиля по винкоду;
    2. Выдать список запчастей по заданной модели авто;
    3. Выдать список автомобилей, которые в данный момент находятся в сервисе;
  1. **Особенности моделирования БД для АС(название возможно другое)**

Опираясь на описание предметной области, бизнес-процессов и запросов появляется возможность выделить сущности в информационной системе, а также их атрибуты (таблица 1)

ТАБЛИЦА НЕМНОГО ПОМЕНЯЕТСЯ, Т.К СЮДА ЕЩЕ ДОБАВИТСЯ ОДНА ТАБЛИЦА «МАРКА»

Таблица 1 - Сущности и атрибуты

|  |  |
| --- | --- |
| Сущность | Атрибуты |
| Автомобиль | ID Автомобиля  Модель  Год  Номер СТС  ID Владельца |
| Владелец Авто | ID Владельца  ФИО  Телефон  Адрес  Номер водительского удостоверения |
| Сотрудник | ID Сотрудника  ФИО  Должность  Адрес проживания  Номер мобильного телефона  Паспорт |
| Заказ | ID Заказа  Дата заказа  Статус заказа  Описание проблемы  ID Автомобиля  Сумма оплаты |
| Проданные запчасти в рамках заказа | ID Проданные запчасти  ID Заказа  Статус  Дата изменения статуса  ID Запчасти  Количество |
| Услуги в рамках заказа | ID Истории заказа  Статус  Дата изменения статуса  ID Сотрудника  ID Услуги |

Можно еще что-то вписать в этот раздел, в конце должна быть диаграмма, ее надо малеха подправить.

* 1. **Механизмы защиты данных в СУБД PostgreSQL**

**ПОПРОСИ ГПТ ПЕРЕФРАЗИРОВАТЬ ЭТОТ КУСОК!!! (у меня просто нет платной версии((((((((((((((((((((((((((( и местами методы поменяй как захочешь**

**Аутентификация и авторизация:**

PostgreSQL поддерживает несколько методов аутентификации, включая пароль, MD5-хешированный пароль, Kerberos, LDAP и SSL-сертификаты.

Для каждой базы данных в PostgreSQL можно настроить различные уровни доступа и привилегии для пользователей и ролей. Это помогает контролировать доступ к данным.

**Резервное копирование и восстановление:**

PostgreSQL предоставляет механизмы резервного копирования и восстановления данных. Вы можете создавать регулярные резервные копии баз данных, чтобы защитить данные от потери или повреждения.

Для резервного копирования и восстановления данных в PostgreSQL используются инструменты, такие как pg\_dump и pg\_restore.

**Журналирование и аудит:**

PostgreSQL поддерживает журналирование операций, которое записывает все изменения данных и запросов, выполняемых в базе данных. Это позволяет восстановить базу данных в случае сбоя или ошибки.

Механизмы аудита PostgreSQL позволяют отслеживать и регистрировать действия пользователей, обновление данных и другие события, связанные с базой данных.

**Управление доступом и RLS политика**:

Использование механизмов управления доступом и механизма разграничения доступа на уровне строк для ограничения прав доступа пользователей к определенным данным и функционалу. Применение политик доступа на основе ролей и групп пользователей.

**Обновления и патчи:**

Команда разработчиков PostgreSQL постоянно работает над обновлениями и патчами для устранения уязвимостей безопасности и повышения общей безопасности системы. Регулярное обновление PostgreSQL до последней версии поможет удержать защиту данных.

**Маскировка:**

Представляет собой механизм, который обеспечивает обработку и защиту конфиденциальных данных, предоставляя замену реальных значений в определенных столбцах или полях таблицы. Это полезно, когда нужно скрыть или замаскировать конфиденциальные данные, такие как социальные номера, номера кредитных карт или адреса электронной почты.

Маскировка может быть реализована с помощью различных методов в PostgreSQL, включая использование функций, правил или псевдонимов. Например, вы можете создать функцию, которая заменяет часть строки или значение поля на "\*\*\*\*" или другой символ маскировки.

**Триггеры**:

Это хранимые процедуры, которые выполняются автоматически при определенных событиях в таблице, таких как вставка, обновление или удаление записей. Они позволяют выполнять пользовательские операции, воздействуя на данные, сохраняя целостность базы данных и обеспечивая дополнительную функциональность.

**ШИФРОВАНИЕ ДАННЫХ (тебе это подробно) (примеров чет не смог найти в инете прям понятных):**

PostgreSQL поддерживает шифрование данных в покое, используя технологии, такие как SSL/TLS. Это защищает данные от несанкционированного доступа при передаче по сети.

Для шифрования данных в базе данных PostgreSQL можно использовать модуль pgcrypto, который предоставляет различные функции шифрования.

[**https://postgrespro.ru/docs/postgresql/15/encryption-options**](https://postgrespro.ru/docs/postgresql/15/encryption-options)

**Крч Дим, вот на этом сайте хорошо написано про шифрование, но работы будут проверять на антиплагиат, поэтому скопипастить тупо не получится, копируй кусками текст по абзацам и проси ГПТ перефразировать, это пару минут времени, мне тупо негде это сделать, если к своей меня подключишь, то я утром сделаю!!!**

[**https://postgrespro.ru/docs/postgresql/14/pgcrypto**](https://postgrespro.ru/docs/postgresql/14/pgcrypto) **тут тоже про шифрование, но в тех плане.**

**РОЛЕВАЯ ПОЛИТИКА (тебе это подробно):**

[**https://sql-ex.com/blogs/?/Osnovy\_PostgreSQL\_roli\_i\_privilegii.html**](https://sql-ex.com/blogs/?/Osnovy_PostgreSQL_roli_i_privilegii.html)

[**https://postgrespro.ru/docs/postgresql/15/user-manag**](https://postgrespro.ru/docs/postgresql/15/user-manag) **вот тут все про роли, тут прям от А-Я написано, я хз что ты отсюда захочешь в отчет вставить**

**Я тут вот не понял, тебе надо описать просто ее в постгри или то, как она у тебя реализована в проге. Но склоняюсь к первому варианту, поэтому небольшой текст про рлс, далее можно с сайта выше накидать примеров по созданию ролей, присасыванию к ним юзера.**

**И про примеры, хз какие сюда примеры воткнуть можно ибо они должны быть в контексте БД, если только наши сюда.**

Ролевая политика (Role-based Access Control, RBAC) в PostgreSQL предоставляет мощный механизм управления доступом и безопасностью баз данных. Она позволяет администраторам определять роли и назначать им различные привилегии и ограничения.

В PostgreSQL роли могут быть созданы для пользователей или групп пользователей. Роли могут иметь различные привилегии, такие как создание объектов, чтение, запись и удаление данных. Каждая роль может быть назначена одной или нескольким базам данных, и привилегии можно назначать как на уровне базы данных, так и на уровне схем или объектов.

Ролевая политика PostgreSQL основана на концепции наследования ролей, что означает, что определенные привилегии могут быть унаследованы от других ролей. Например, если у роли A есть определенные привилегии, то роль B, которая унаследует от роли A, также будет иметь эти привилегии.

С помощью ролевой политики можно также определить ограничения доступа на уровне строк. Например, можно настроить так, чтобы пользователь мог просматривать, изменять или удалять только определенные строки в таблице, основываясь на определенных условиях.

Важным аспектом ролевой политики является возможность управления доступом к системным таблицам и функциям. Администратор может назначать привилегии на уровне системы для осуществления контроля над созданием и изменением ролей, настройкой параметров безопасности и другим административным задачам.

# Практические аспекты разработки ас/ис «АВТОСЕРВИС» и ее защиты

## **Реализация серверной части**

В процессе установки были загружены следующие компоненты (Рисунок 2.1):

* **PostgreSQL Server** – сам сервер СУБД
* **PgAdmin 4** – визуальный редактор SQL
* **Stack Builder** – дополнительные инструменты для разработки
* **Command Line Tools** – инструменты командной строки

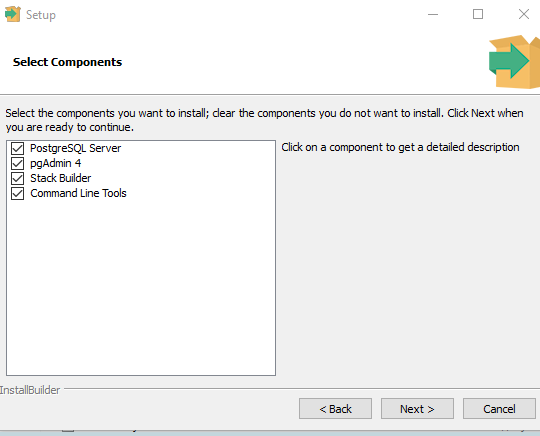


Рисунок 2.1.1 – Выбор компонентов установки

Был установлен пароль для пользователя root, рисунок 2.2.

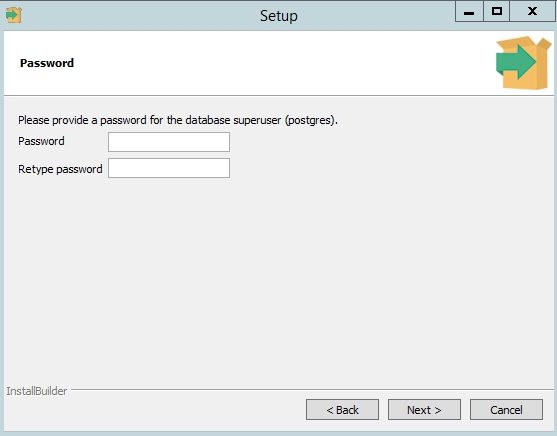


Рисунок 2.1.2 – Установка пароля для пользователя по умолчанию

По умолчание СУБД был настроен на порту 5432, который далее был добавлен в исключения в правилах файрволла.

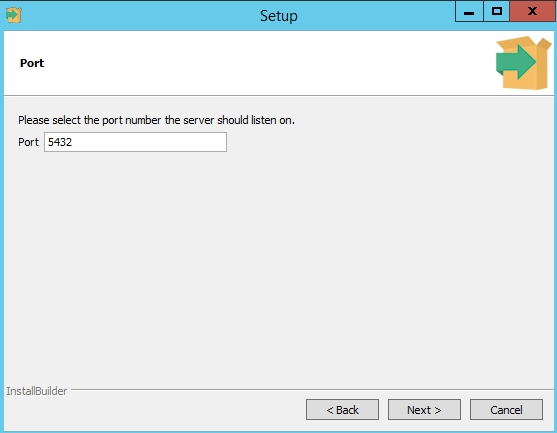


Рисунок 2.1.3 – Настройка порта для СУБД

Рисунок 2.1.4 – Создание БД «Автосервис»

**НЕ ЗНАЮ ПРИГОДИТСЯ ТЕБЕ ИЛИ НЕТ, но тут описал тот рлс, что тебе кидал.**

Для реализация ролевой политики и политики RLS в информационной системе «Автосервис» были выделены следующие роли:

1. **Админ** – роль, включающая в себя главного администратора системы. РИСУНОК

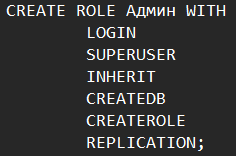


Рисунок 2.2.1 – SQL запрос на создание роли Админ

1. **ГлавМас** – роль, включающая в себя главного мастера РИСУНОК

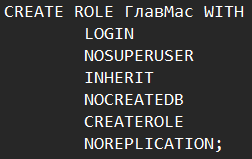


Рисунок 2.2.2 – SQL запрос на создание роли ГлавМас

1. **Сотрудники** – роль, включающая в себя сотрудников РИСУНОК

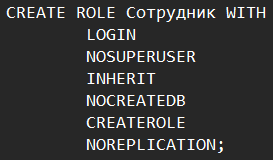
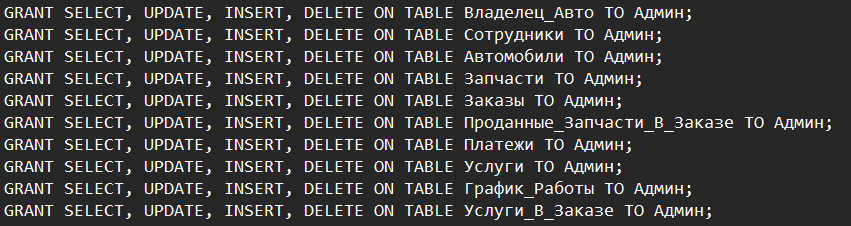


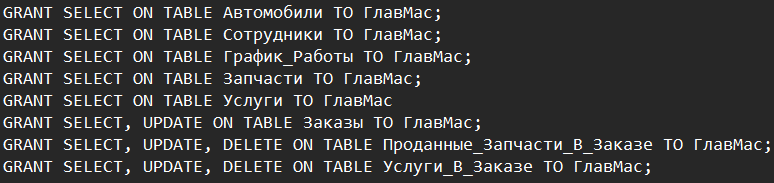
Рисунок 2.2.3 – SQL запрос на создание роли Сотрудники

На основе созданных ролей была организована политика система разрешений в ИС:

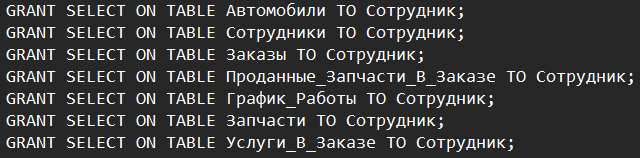
1. Для роли «Админ» были выданы следующие разрешения в соответствующих таблицах РИСУНОК:



1. Участники роль «ГлавМас» смогут работать со следующими таблицами РИСУНОК:

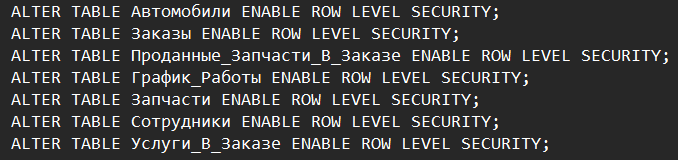


1. Пользователи, относящиеся к роли «Сотрудник» могут видеть таблицы, которые показаны ниже РИСУНОК:

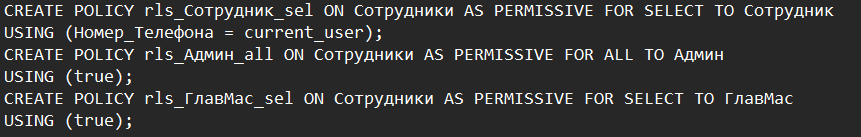


Опираясь на созданные роли, была организована политика RLS изображённая на рисунках ниже:

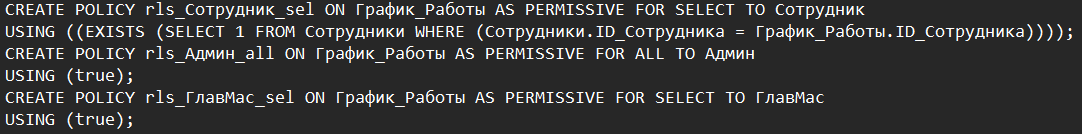
1. Включение политики на уровне строк для таблиц представлено на рисунке .



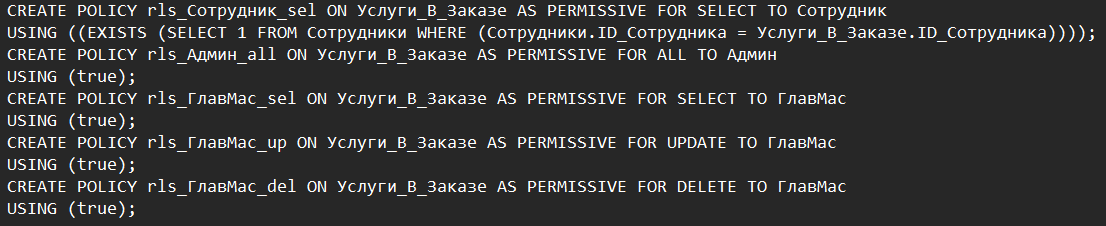
1. Реализация RLS политики для таблицы «Сотрудники» изображена на рисунке .



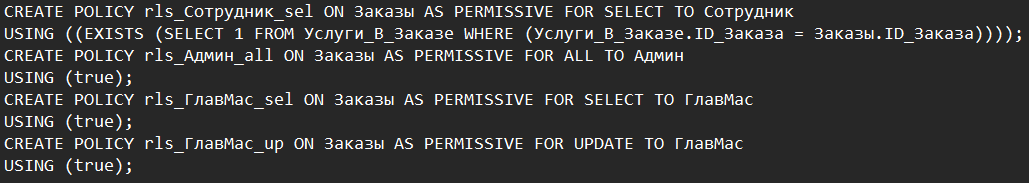
1. Реализация RLS политики для таблицы «График\_работы» изображена на рисунке .



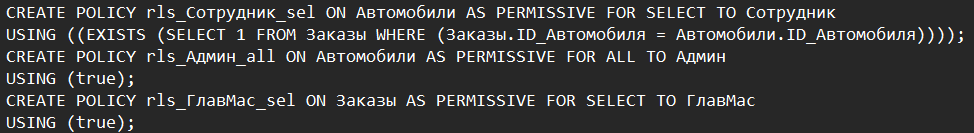
1. Реализация RLS политики для таблицы «Услуги\_В\_Заказе» изображена на рисунке .



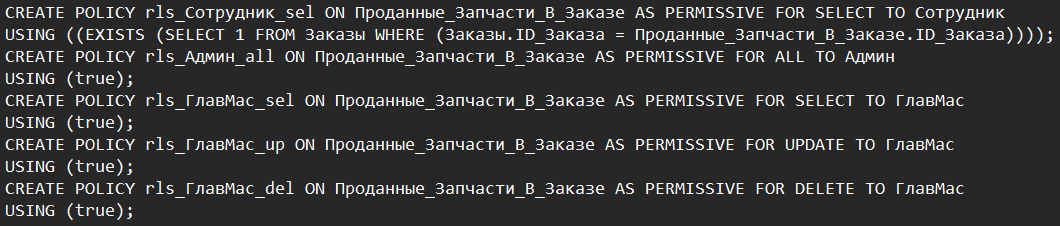
1. Реализация RLS политики для таблицы «Заказы» изображена на рисунке .



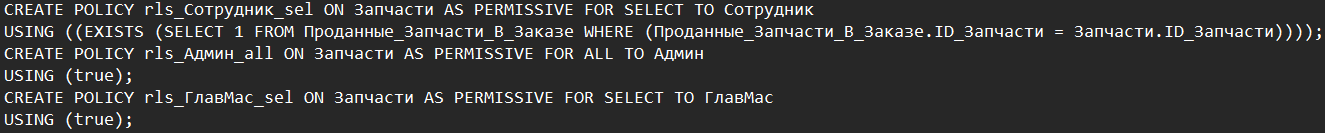
1. Реализация RLS политики для таблицы «Автомобили» изображена на рисунке .



1. Реализация RLS политики для таблицы «Проданные\_Запчасти\_В\_Заказе» изображена на рисунке .



1. Реализация RLS политики для таблицы «Запчасти» изображена на рисунке .



Библиография

<https://habr.com/ru/news/734772/>

<https://www.postgresql.org/docs/>

[https://vitobotta.com/2022/02/02/data-anonymization-with-postgres/ (про](https://vitobotta.com/2022/02/02/data-anonymization-with-postgres/%20(про) маскировку и платный код для винды)

<https://www.cherryservers.com/blog/create-postgresql-stored-procedure>

<https://postgrespro.ru/docs/postgresql/15/xfunc-sql>

<https://docs.python.org/3/index.html>

<https://doc.qt.io/qtforpython-5/contents.html>

<https://pyqt5.files.wordpress.com/2017/06/pyqt5tutorial.pdf>

<https://pgdocs.ru/docs/trigger.html>