

ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ

АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ «ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

ВОЕННЫЙ УЧЕБНЫЙ ЦЕНТР

РЕФЕРАТ

**Тема: «Предназначение, состав и особенности реализации механизмов защиты , встроенных в СУБД»**

**Выполнил:**

Студент 1810 взвода

Ю.Л. Попов

17 мая 2020 г.

**Проверил:**

преподаватель военно-учебного центра

А. П. Поляков

«\_\_» \_\_\_\_\_\_\_ 2020 г.

Москва 2020

Содержание

[Введение 3](#__RefHeading___Toc371_1084908452)

[Немного о СУБД 4](#__RefHeading___Toc483_1084908452)

[Средства обеспечения безопасности доступа, используемые в известных СУБД 7](#__RefHeading___Toc589_1084908452)

[Почему невозможно обеспечить информацию только средствами ОС? 8](#__RefHeading___Toc375_1084908452)

[SQL Server 10](#__RefHeading___Toc147_1584526686)

[Пользователи 10](#__RefHeading___Toc149_1584526686)

[Создание роли 13](#__RefHeading___Toc151_1584526686)

[Права доступа 14](#__RefHeading___Toc157_1584526686)

[Права на таблицы 17](#__RefHeading___Toc159_1584526686)

# Введение

Основной целью данной работы является введение в само понятие СУБД, обзор реализаций механизмов защиты встроенных в СУБД и получение ответа на вопрос: «Почему для обеспечения безопасности СУБД недостаточно средств защиты ОС?»

# Немного о СУБД

Давайте разберёмся с терминами и понятиями баз данных:

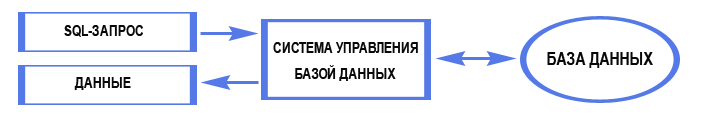
***База данных*** - набор сведений, хранящихся некоторым упорядоченным способом.

Можно сравнить базу данных со шкафом, в котором хранятся документы. Иными словами, база данных - это хранилище данных. Сами по себе базы данных не представляли бы интереса, если бы не было систем управления базами данных (СУБД).

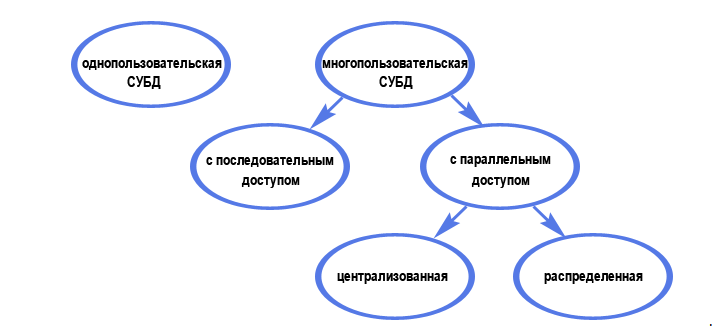
***Система управления базами данных(СУБД)*** - это совокупность языковых и программных средств, которая осуществляет доступ к данным, позволяет их создавать, менять и удалять, обеспечивает безопасность данных и т. д.

В общем СУБД - это система, позволяющая создавать базы данных и манипулировать сведениями из них. А осуществляет этот доступ к данным СУБД посредством специального языка — SQL.

***SQL*** - язык структурированных запросов, основной задачей которого является предоставление простого способа считывания и записи информации в базу данных.  
  
Итак, простейшая схема работы с базой данных выглядит примерно так:



По характеру использования СУБД делят на однопользовательские (предназначенные для создания и использования БД на персональном компьютере) и многопользовательские (предназначенные для работы с единой БД нескольких компьютеров, объединённых в локальные сети).



Нас интересуют многопользовательские БД. Самый распространённый тип БД — реляционной

Именно о нем мы и поговорим

Реляционные базы данных состоят из таблиц. Каждая таблица состоит из столбцов и строк Таблицы в реляционных базах данных обладают рядом свойств. Основными являются следующие:

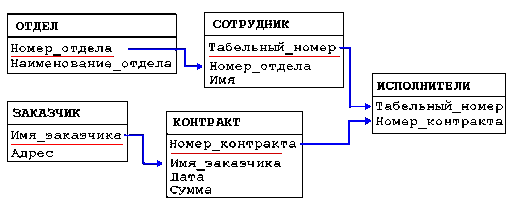
* В таблице не может быть двух одинаковых строк.
* Столбцы располагаются в определённом порядке, который создаётся при создании таблицы. В таблице может не быть ни одной строки, но обязательно должен быть хотя бы один столбец.
* У каждого столбца есть уникальное имя (в пределах таблицы), и все значения в одном столбце имеют один тип (число, текст, дата...).
* На пересечении каждого столбца и строки может находиться только значение

Рассмотрим, как хранится в памяти простейшая БД:

Как уже говорилось раньше, БД — это набор таблиц и связи между ними. Например, нам нужно хранить информацию об исполнителе: табельный номер и номер контракта. Но нам также необходимо знать, что это за сотрудник и в каком отделе он работает. Вопрос как же тогда хранить столько связей?

Для этого придумали создавать несколько таблиц, где каждая ячейка может указывать на ещё одну таблицу. Итак, у нас итоговая таблица по исполнителю, которая хранит его табельный номер и номер контракта. Если мы хотим узнать более подробно, про табельный номер, мы можем перейти в таблицу, где такой номер есть и посмотреть кому он принадлежит. В нашем случае, мы попадём в таблицу «Сотрудник», и узнаем его имя и номер отдела. А если в нашей организации много отделов и мы не помним у какого отдела какой номер? Не проблема! Для этого сделаем ещё одну таблицу «Отдел», в котором будет сопоставляться номер отдела и его наименование.

Таким образом, наша БД, представляет собой множество таблиц, и ссылки, по которым мы можем переходить и получать более подробную информацию.



# Средства обеспечения безопасности доступа, используемые в известных СУБД

Существенным аспектом СУБД является защита данных. В современных СУБД поддерживается как избирательный, так и обязательный подходы к обеспечению безопасности данных. В случае избирательного управления, некий пользователь обладает различными правами, или привилегиями, и полномочиями при работе с различными объектами. В случае обязательного управления, каждому объекту присваивается некий квалификационный уровень, а каждому пользователю предоставляются права доступа к тому или иному уровню.

Как правило, для идентификации и проверки подлинности пользователя применяется либо соответствующий механизм операционной системы, либо функциональные возможности SQL-оператора «connect».

В момент начала сеанса работы с сервером базы данных пользователь идентифицируется под своим логином, а средством аутентификации служит пароль. Все операции над базой данных, которые будут выполнены после этого, СУБД связывает с конкретным пользователем, который запустил приложение.

Одна из проблем защиты данных возникает по той причине, что при работе с базами данных инициируются процессы, которые могут запускаться как пользователями, так и прикладными программами. Одно из решений проблемы заключается в том, чтобы прикладной программе также были предоставлены некоторые привилегии доступа к объектам базы данных. В этом случае пользователь, не обладающий специальными привилегиями доступа к некоторым объектам базы данных, может запустить прикладную программу, которая имеет такие же привилегии. В СУБД «Oracle» решение данной проблемы обеспечивается использованием механизма ролей. Роль представляет собой именованный объект, хранящийся в базе данных, и связывается с конкретной прикладной программой для придания последней привилегий доступа к базам данных, таблицам, представлениям и процедурам базы данных. Роль создаётся и удаляется администратором базы данных, ей может быть придан определённый пароль. Как только роль создана, ей можно предоставить привилегии доступа к объектам базы данных.

Ещё одним эффективным способом обеспечения безопасности является шифрование данных.

Современные СУБД включают резервное копирование и аудит как непременные составляющие системы безопасности.

# Почему невозможно обеспечить информацию только средствами ОС?

Чтобы обеспечить безопасность СУБД, для начала нужно ответить на вопрос: кто в компании вообще пользуется базой данных и имеет к ней доступ?

Существует три большие группы пользователей СУБД, которых условно можно назвать **операторами; аналитиками; администраторами**.

* *Операторы - это*  те, кто либо заполняет базу данных (вводят туда вручную информацию о клиентах, товарах и т. п.), либо выполняют задачи, связанные с обработкой информации: кладовщики, отмечающие перемещение товара, продавцы, выписывающие счета и т. п.
* *Аналитики — это*  те, ради кого, собственно, создаётся эта база данных: логистики, маркетологи, финансовые аналитики и прочие. Эти люди по роду своей работы получают обширнейшие отчеты по собранной информации.
* Администраторы - Эта категория людей может только в общих чертах представлять, что хранится и обрабатывается в хранилище данных. Но они решают ряд важнейших задач, связанных с жизнеобеспечением системы, ее отказо- и катастрофоустойчивости.

Но различные роли по отношению к обрабатываемой информации - не единственный критерий. Указанные группы различаются еще и по способу взаимодействия с СУБД

Операторы чаще всего работают с информацией через различные приложения. Для данные группы пользователей безопасность необходимо обеспечивать встроенными средствами СУБД. И чаще всего о безопасности СУБД судят только по этому критерию. Но это неправильно! Беда в том, что основные проблемы с безопасностью приходятся на две оставшиеся категории пользователей.

Проблема с аналитиками заключается в том, что они работают с СУБД на уровне ядра. Они должны иметь возможность задавать и получать всевозможные выборки информации из всех хранящихся там таблиц. Вот тут как раз необходимы средства защиты ОС!

С администраторами дело обстоит ненамного лучше. Начиная с того, что в крупных информационных системах их число сопоставимо с числом аналитиков. И хотя абсолютно полными правами на СУБД наделены лишь два-три человека, администраторы, решающие узкие проблемы (например резервное копирование данных), все равно имеют доступ ко всей информации, хранящейся в СУБД. А для обеспечения безопасности от этой группы пользователей необходим комплексный подход: средства СУБД + средства ОС

БД отличается от обычного файла на локальном компьютере тем, что БД хранится на удалённом сервере и каждый, желающих получить доступ к БД должен подключиться к этому серверу.

**Итог:**

Средствами ОС можно обеспечить доступ пользователей к самой БД, а уже в самой БД доступ к различным таблицам обеспечивается средствами СУБД.

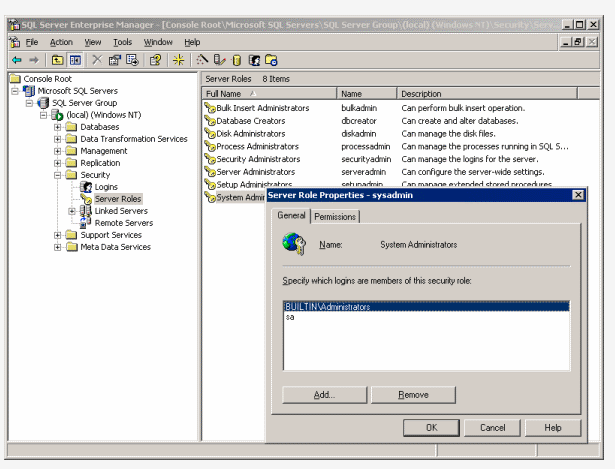
Пример настройки SQL Server мы рассмотрим ниже

# SQL Server

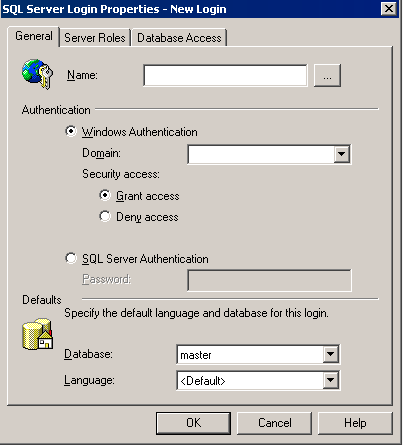
Рассмотрим, как происходит настройка безопасности БД в приложение SQL Server. Это приложение предоставляет графический интерфейс SQL команадам.

## Пользователи

Для управления пользователями откроем в Enterprise Manager ветку Security/Logins. В правой части появится список всех пользователей сервера. По умолчанию доступ имеют администраторы домена и встроенная учетная запись sa



Для добавления нового пользователя, щёлкнем правой кнопкой в пустом месте правой половины окна и в появившемся меню выбери New login. Перед нами откроется окно добавления нового пользователя. В самом верху окна выбирается имя пользователя. Если нужно выбрать уже существующего юзера домена или компьютера, то щёлкнем по кнопке (…) справа от поля ввода, и мы увидим окно поиска пользователя в домене



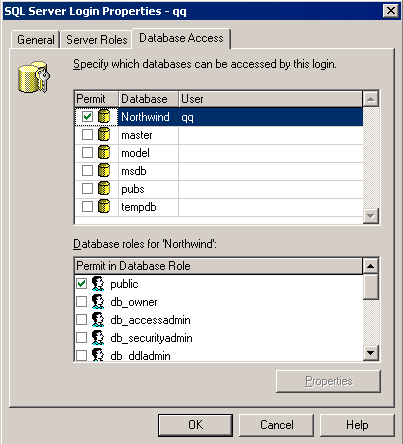
Чуть ниже выбирается тип аутентификации – Windows или SQL Server.

Если выбрать Windows, то пароль указывать не надо, потому что сервер сам возьмёт его из системы. Но зато можно выбрать один из переключателей – Grant access (разрешить доступ) или Deny access (запретить). Во втором случае пользователь будет прописан в базе, но подключиться он не сможет – запрещено.

Если выбрать аутентификацию SQL Server, то нужно будет задать пароль, потому что в этом случае, он будет храниться в системных таблицах сервера баз данных. Даже если в настройках сервера указана только аутентификация Windows, записи SQL сервера создавать разрешено, но вот войти в систему с этими записями будет невозможно

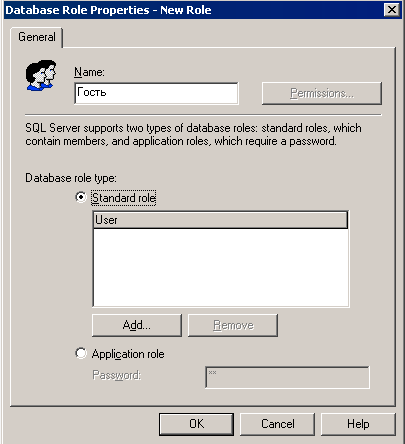
На закладке Server Roles можно указать, какой серверной роли будет принадлежать юзер. Таким образом, уже на этапе создания можно включить юзеров в нужные роли. На закладке Database Access указываем базы, с которыми может работать юзер. Здесь окно разделено на две части: в верхней половине можно выбирать базу данных, к которой разрешён доступ, а в нижнем списке выбирается роль базы данных. В зависимости от выбранной роли в базе, пользователю будут доступны те или иные права. Один пользователь может входить в несколько ролей

Давай для примера создадим учётную запись qq, которой будет разрешён доступ к базе данных Northwind. Это стандартная тестовая база данных, которая создаётся при установке сервера



## Создание роли

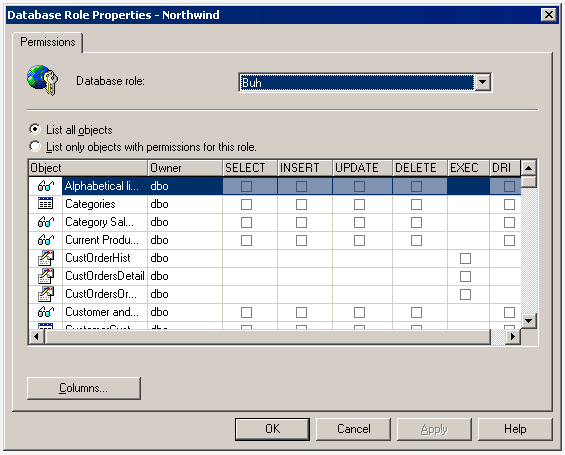
Для создания новой роли базы данных щёлкнем правой кнопкой по ветке Databases/Имя базы/Roles и в появившемся меню выбери пункт New database role. Перед нами открывается окно создания новой роли. В самом верху окна нужно ввести имя роли. Например, мы хотим создать роль для бухгалтеров фирмы. Для этого введем имя Buh



Чуть ниже выбирается тип роли. Мы остановим стандартную, которая выбрана по умолчанию. При этом в центре окна есть список пользователей, которые будут входить в роль. Сейчас список пуст, но если нажать кнопку Add, то можно добавить пользователей. Добавим для примера созданного ранее qq. Больше ничего сделать на этапе создания роли нельзя. Сохраняй изменения нажатием OK

## Права доступа

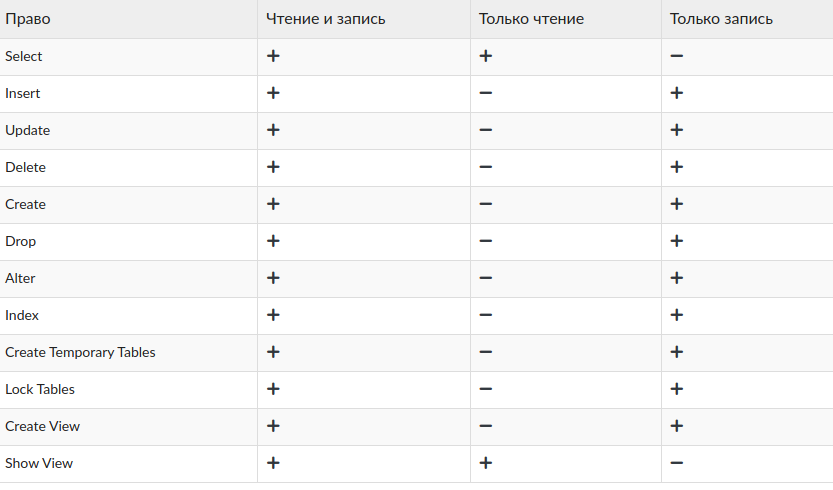
Теперь посмотрим, как можно назначать права доступа. Дважды щелкнем по созданной ранее роли buh и снова откроется окно, которое было при создании, но в этот раз оно открылось для редактирования. Обратим внимание, что кнопка Permission стала доступной, чего не было раньше. Только когда роль уже прописана в базе, можно изменять ее права. Щелкаем по этой кнопке и видим окно настройка прав на объекты базы данных



Вверху окна находится список ролей базы, чтобы можно было быстро переключаться между ними, а сейчас там выбрана роль Buh. В центре окна находится большая сетка из следующих колонок:

* Object имена объектов;
* Owner владелец объекта;
* SELECT разрешение на просмотр данных или выполнение команды SELECT.
* INSERT разрешение на добавление данных или выполнение команды INSERT.
* UPDATE разрешение на изменение данных или выполнение команды UPDATE
* DELETE разрешение на удаление данных или выполнение команды DELETE.
* EXEC разрешение на выполнение хранимых процедур и функций.
* DRI (declarative referential integrity) обеспечение целостности.

Ниже, представлена таблица прав для каждой роли(Более удобный вид, написанного выше):

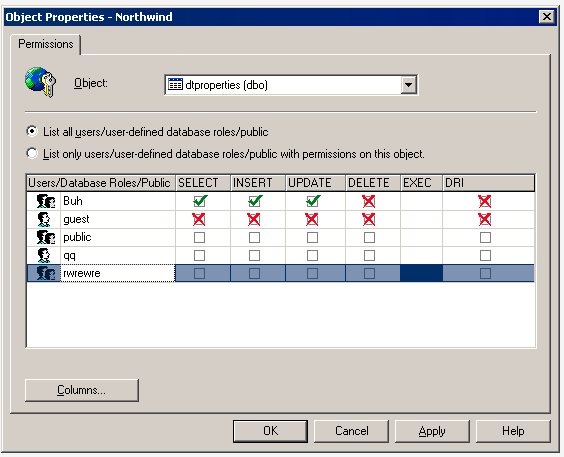


В новой роли никаких прав нет. Чтобы добавить возможность просмотра таблицы, например, Categories, нужно щёлкнуть в квадрате на пересечении строки и Categories и колонки SELECT. Щелчок устанавливает в этом квадрате зелёную галочку, что соответствует разрешению. Второй щелчок меняет галочку на красный крест, что соответствует запрету.

Это бывает необходимо, если пользователь может получить доступ, если он находиться одновременно и в другой роли, где доступ на выбранное действие разрешен. Третий щелчок снимает какие-либо разрешения на действие и оставляет квадрат пустым. Это означает, что доступа нет, но он может быть делегирован, если пользователь участвует в другой роли с разрешёнными правами на объект или права указаны явно.

## Права на таблицы

Давай теперь посмотрим, как можно давать права на определенные объекты. Для начала посмотрим таблицы. Выберем в дереве объектов ветку Databases/Northwind/Tables и в правой части будет показан список всех таблиц. Щелкнем по любой таблице правой кнопкой и в появившемся меню выбери All tasks/Manage permissions. Перед нами открывается окно настройки прав. Окно похоже на распределение прав ролей, только вместо списка объектов стоит список пользователей. Объект и так ясен – это та таблица, по которой мы щелкали, и ее имя виднеться в выпадающем списке вверху окна. Теперь нам остается только указать права для этого объекта различным пользователям.



Итого, мы рассмотрели создания пользователя, добавление его в группы и предоставление доступа к определённой таблице с определёнными правами