Министерство образования и науки Российской Федерации

Калужский филиал
федерального государственного бюджетного образовательного
учреждения высшего образования
**«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»**
(КФ МГТУ им. Н.Э. Баумана)

С.А. Глебов

ОБЕСПЕЧЕНИЕ МНОГОПОЛЬЗОВАТЕЛЬСКОГО РЕЖИМА РАБОТЫ

Методические указания по выполнению лабораторной работы
по курсу «Базы данных»

Калуга – 2018

УДК 004.65
ББК 32.972.134
Г53

Методические указания составлены в соответствии с учебным планом КФ МГТУ им. Н.Э. Баумана по направлению подготовки 09.03.04 «Программная инженерия» кафедры «Программного обеспечения ЭВМ, информационных технологий и прикладной математики».

Методические указания рассмотрены и одобрены:

- Кафедрой «Программного обеспечения ЭВМ, информационных технологий и прикладной математики» (ФН1-КФ) протокол № 7 от «21» февраля 2018 г.

И.о. зав. кафедрой ФН1-КФ _____ к.т.н., доцент Ю.Е. Гагарин

- Методической комиссией факультета ФНК протокол № 2 от «28» 02 2018 г.

Председатель методической комиссии факультета ФНК _____ к.х.н., доцент К.Л. Анфилов

- Методической комиссией КФ МГТУ им.Н.Э. Баумана протокол № 2 от «06» 03 2018 г.

Председатель методической комиссии КФ МГТУ им.Н.Э. Баумана _____ д.э.н., профессор О.Л. Перерва

Рецензент:
к.т.н., доцент кафедры ЭИУ6-КФ _____ А.Б. Лачихина

Авторы
к.ф.-м.н., доцент кафедры ФН1-КФ _____ С.А. Глебов

Аннотация

Методические указания по выполнению лабораторной работы по курсу «Базы данных» содержат руководство по созданию пользователей и ролей в базах данных, а также назначению им прав доступа к данным и задание на выполнение лабораторной работы.

Предназначены для студентов 3-го курса бакалавриата КФ МГТУ им. Н.Э. Баумана, обучающихся по направлению подготовки 09.03.04 «Программная инженерия».

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Настоящие методические указания составлены в соответствии с программой проведения лабораторных работ по курсу «Базы данных» на кафедре «Программное обеспечение ЭВМ, информационные технологии и прикладная математика» факультета фундаментальных наук Калужского филиала МГТУ им. Н.Э. Баумана.

Методические указания, ориентированные на студентов 3-го курса направления подготовки 09.03.04 «Программная инженерия», содержат руководство по созданию пользователей и ролей в базах данных, а также назначению им прав доступа к данным.

ЦЕЛЬ И ЗАДАЧИ РАБОТЫ, ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ЕЕ ВЫПОЛНЕНИЯ

Целью выполнения лабораторной работы является сформировать практические навыки использования операторов раздачи и аннулирования привилегий.

Основными задачами выполнения лабораторной работы являются:

- Научиться создавать учетные записи и роли
- Научиться назначать привилегии пользователям и ролям

Результатами работы являются:

- Созданные роли и пользователи с назначенными правами
- Подготовленный отчет.

ЗАЩИТА БАЗ ДАННЫХ

Данные являются самым ценным компонентом среды СУБД. Их стоимость может во много раз превышать стоимость аппаратного и программного обеспечения вместе взятых. Поэтому в любой многопользовательской среде (в том числе и в архитектуре клиент-сервер) безопасность данных является одним из важных вопросов.

С ростом значения информации в современном обществе и с ростом числа пользователей баз данных вопросы защиты данных представляют собой одно из приоритетных направлений в информационных технологиях.

Никакая система защиты не дает 100% гарантии. Любой взлом — это лишь вопрос времени и ресурсов. Смысл взлома системы безопасности отпадает, если его стоимость превышает стоимость похищенных данных. Со стороны жертвы следует помнить о том, что стоимость системы защиты не должна превышать возможный ущерб от потери информации.

Вопросы защиты информации включают в себя как проблемы защиты данных, как таковых от несанкционированного доступа (внутренняя защита), так и защиты носителей, аппаратных и программных средств от различных угроз.

Угрозы могут быть *случайные* и *преднамеренные*. Сложнее всего противостоять именно случайным угрозам, в силу их непредсказуемости. Преднамеренные угрозы можно прогнозировать и строить соответствующую защиту.

Угрозы бывают *активные* (целью которых является нарушение нормального функционирования путем воздействия на аппаратные, программные или информационные ресурсы) и *пассивные* (не оказывающие влияния на нормальное функционирование информационных систем и связаны прежде всего с потерей конфиденциальности информации).

Т.о. владельца информации подстерегают две неприятности: утрата (в т.ч. и искажение) данных и потеря их конфиденциальности (кража информации).

Утрата данных понятие очень широкое. Утрата может произойти вследствие:

- стихийных и техногенных катастроф, радиационных воздействий;
- кража оборудования;
- разрушения файловой структуры диска (вирусы, физическая порча);
- удаления файлов БД (случайное и преднамеренное);
- удаления записей, таблиц и др. в самой БД.

Защита БД от угроз со стороны внешней среды (все перечисленные кроме последней) это, фактически защита носителей информации. Они входят в задачи службы безопасности, системного и сетевого администрирования, прочих служб и подразделений.

Вероятность возникновения перечисленных событий никогда не бывает равна нулю, поэтому для минимизации потерь рекомендация одна: страховочные копии БД должны делаться как можно чаще и хранится на другом диске, в другом компьютере, другом здании или даже городе.

В данной лекции будут рассмотрены вопросы безопасности данных внутри самой базы. Эти вопросы решает администратор базы данных.

Внутри базы данных все действия совершаются от имени пользователя (или процедуры). Поэтому каждый пользователь БД должен быть идентифицирован. Конечно, возможна политика тотального доверия, когда все пользователи работают под одной учетной записью (или вовсе без нее), но о безопасности и конфиденциальности данных, в этом случае, речи быть не может.

Идентификация пользователя – еще не гарантия безопасности, а лишь средство обнаружения источника ее нарушения.

Существует три варианта идентификации пользователя:

- по чему-либо, что знает только сам пользователь (пароль, алгоритм, ответы на вопросы);

– по чему-либо, чем он обладает (пластиковая карточка, электронный ключ);
– по физическим параметрам (сетчатка глаза, отпечаток пальца, идентификация голоса и т.п.)

Системы идентификации не совершенны. Пароль – самый дешевый, самый распространенный и самый уязвимый способ идентификации пользователя.

Идентификация пользователя происходит путем назначения ему учетной записи и сравнения с ней данных, вводимых пользователем при авторизации. Никакие действия с базой данных невозможны без предварительной идентификации пользователя по его учетной записи.

В IB/FB имеется изначальная учетная запись SYSDBA (SYStem DataBase Administrator). Пароль — masterkey. Настоятельная рекомендация — поменять пароль после установки сервера. Создание новых учетных записей возможно только пользователем SYSDBA. Учетная запись SYSDBA не имеет никаких ограничений по действиям над базой данных.

Учетные записи хранятся в закодированном виде в системной таблице на уровне сервера. Каждая учетная запись состоит из: имя пользователя в БД (31 символ), пароль (32 символа / 8 значащих), ФИО пользователя.

То, что все учетные записи определяются и хранятся на уровне сервера, позволяет иметь единое пространство пользователей для всех баз данных, находящихся под управлением данного сервера. Однако, это предъявляет повышенные требования к защите самого файла БД, т.к. скопировав файл и подключившись к нему под управлением другого сервера (на котором известен пароль SYSDBA) можно получить полный доступ ко все объектам БД. Основная рекомендация: к файлам БД не должен быть открыт общий доступ (shared access), т.к. в системах клиент-сервер пользователи взаимодействуют не с файлами БД, а с SQL-сервером.

В каждой конкретной БД каждому пользователю назначаются свои права для объектов именно этой БД.

Правило: пользователь не должен иметь доступ к объектам БД, которые не входят в сферу его профессиональных обязанностей.

Перечень разрешенных пользователю операций с конкретным объектом называется правами или привилегиями пользователя. Помимо пользователей, правами обладают также роли, хранимые процедуры, триггеры и просмотры.

Сами права бывают на: таблицы, просмотры и хранимые процедуры.

НАЗНАЧЕНИЕ ПРАВ ДОСТУПА К БАЗЕ ДАННЫХ

На таблицы и просмотры существуют права: SELECT, INSERT, UPDATE, DELETE и REFERENCES (право создавать ограничения внешнего ключа в других таблицах, ссылающихся на данную таблицу)

На отдельные столбцы таблицы: UPDATE и REFERENCES

На хранимые процедуры: EXECUTE (право выполнять процедуру)

Перед выполнением запроса пользователя сервер выясняет его права на объекты запроса, в зависимости от этого, выполняет запрос или отказывает в его выполнении. Для пользователя SYSDBA, такая проверка не выполняется в принципе, поэтому результат запроса получается чуть быстрее.

В отличие от учетных записей (которые не принадлежат конкретной БД и, поэтому не могут создаваться при помощи DDL), привилегии учетных записей — определяются на уровне каждой БД операторами DDL: GRANT (предоставить) и REVOKE (отозвать).

Общий синтаксис для предоставления привилегий:

GRANT <список привилегий> | ALL | <роль>
ON <таблица> | <просмотр> | <ХП> | <роль>
TO <список пользователей> | PUBLIC | <роль> | <триггер> | <ХП>
[WITH GRANT OPTION]
[WITH ADMIN OPTION]
<привилегия> = SELECT |
INSERT |
DELETE |
UPDATE [(<список столбцов>)] |
REFERENCE [(<список столбцов>)] |
EXECUTE

ALL — означает пакет привилегий Select & Insert & Update & Delete, исключая EXECUTE.

PUBLIC — означает всех пользователей БД, исключая просмотры, триггеры, ХП и роли.

Опция WITH GRANT OPTION — дает возможность пользователю, получившему привилегии, самому раздавать эти привилегии.

Опция WITH ADMIN OPTION — дает возможность пользователю, получившему членство в роли, давать это членство другим пользователям.

GRANT SELECT
ON Movie
TO User1, User2
WITH GRANT OPTION

Теперь пользователи User1, User2 не только обладают правом чтения таблицы Movie, но и могут передать это право другим пользователям.

Когда появляется новая учетная запись у нее нет никаких привилегий на уже существующие объекты. Однако у нее есть возможность создавать собственные объекты. Создатель или владелец объекта (owner) автоматически обладает всеми правами на созданный объект, в том числе и правом передачи прав (WITH GRANT OPTION).

Необходимо помнить, что при создании объектов, использующих уже существующие объекты, владелец создаваемого объекта должен обладать соответствующими привилегиями к существующим.

Например, имеется таблица:

CREATE TABLE Table1 (A INTEGER, B INTEGER);

Чтобы создать просмотр:

CREATE VIEW View1 (C)
AS SELECT A FROM Table1

у создателя просмотра должно быть право SELECT к таблице Table1.

Просмотр является обновляемым и чтобы успешно выполнить оператор:

INSERT INTO View1 VALUES (123)

У пользователя должна быть привилегия INSERT либо на просмотр View1, либо на таблицу Table1.

Если пользователь запускает хранимую процедуру (или в результате действий пользователя запускается триггер), то все действия, выполняемые ХП или триггером, выполняются от имени и с учетом привилегий пользователя.

Но хранимые процедуры и триггеры могут обладать и своими привилегиями Select | Insert | Update | Delete | Execute.

Пользователь вообще не обладающий этими правами, но имеющий право EXECUTE к такой процедуре, может через ее посредничество модифицировать данные.

Это вполне реальный прием, позволяющий скрыть таблицы от пользователей, предоставив им только возможность запускать процедуры.

Политика журнализации, например, предполагает, что при изменении данных сработает триггер, который занесет запись в log-таблицу. Значит, у этого пользователя должна быть привилегия Insert на log-таблицу. А это означает, что пользователь может добавлять в нее записи минуя триггер. Что не является правильным.

Поэтому правом вставки записей должен обладать триггер, а не пользователь.

Очевидно, что привилегии пользователя могут быть достаточно сложными. И при большом количестве таблиц процесс выдачи прав будет очень трудоемким.

Для упрощения администрирования введены роли (role). Роль — это предварительно созданный и именованный набор привилегий. Оптимизация администрирования заключается в том что пользователю можно назначить привилегии роли или другими словами предоставить членство в роли. Пользователи подключаются к БД с использованием привилегий одной из ролей. Для этого пользователи должны обладать правом членства в одной или нескольких ролях.

Роли — объекты уровня базы данных, в отличие от пользователей, которые являются объектами уровня сервера.

Для создания роли используется оператор DDL:

CREATE ROLE <role_name> Например, CREATE ROLE Reader

Данная роль будет иметь только привилегию SELECT на все таблицы:

GRANT SELECT ON Movie TO Reader;

GRANT SELECT ON Prod TO Reader;

GRANT SELECT ON Star TO Reader;

GRANT SELECT ON StarIn TO Reader;

Теперь пользователям можно разрешить членство в данной роли:

GRANT Reader TO User1, User2, UserN

Создатель роли имеет право ее администрирования, т.е. управления членством в ней через WITH ADMIN OPTION.

При подключении к БД пользователь имеет возможность указать роль, привилегиями которой он будет пользоваться на время текущего соединения. Даже если у самого пользователя нет никаких прав. Для этого он должен обладать членством в данной роли.

CONNECT ‘localhost:C:\Database\mainbase.gdb’

USER User2 PASSWORD ‘Pass2’

ROLE ‘Reader’

Если же пользователь обладает своими собственными привилегиями, то они перекрываются привилегиями роли.

Если пользователь не имеет членства в роли, то в данном сеансе соединения с БД он будет обладать только своими привилегиями.

Аннулирование привилегий выполняется оператором DDL:

REVOKE <список привилегий> | ALL | <роль> | GRANT OPTION

ON <таблица> | <просмотр> | <ХП> | <роль>

FROM <список пользователей> | PUBLIC | <роль> |<триггер> |<ХП>

При этом перевыданные привилегии (выданные с WITH GRANT OPTION) аннулируются каскадно.

Оператор REVOKE не генерирует исключительных ситуаций.

ЗАДАНИЕ НА ЛАБОРАТОРНУЮ РАБОТУ

В базе данных, созданной в прошлой лабораторной, создать пользователей, назначить им привилегии, создать роли и создать членство пользователей в одной и нескольких ролях. Выполнить подключение к БД различных пользователей и проверить действие привилегий и механизма замены привилегий пользователя привилегиями роли. Проверить каскадную выдачу привилегий и каскадное действие оператора аннулирования привилегий.

КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ

1. Раскройте понятие «активные угрозы».
2. Раскройте понятие «пассивные угрозы».
3. Перечислите причины утраты данных.
4. Перечислите права на таблицы и просмотры.
5. Приведите общий синтаксис предоставления прав.
6. Перечислите права на хранимые процедуры и триггеры.
7. Раскройте понятие «роль».
8. Приведите общий синтаксис создания роли.
9. Приведите общий синтаксис аннулирования привилегий.
10. Приведите общий синтаксис задания членства пользователя в какой-либо роли.

ФОРМА ОТЧЕТА ПО ЛАБОРАТОРНОЙ РАБОТЕ

На выполнение лабораторной работы отводится 2 занятия (4 академических часа: 3 часа на выполнение и сдачу лабораторной работы и 1 час на подготовку отчета).

Номер варианта студенту выдается преподавателем.

Отчет на защиту предоставляется в печатном виде.

Структура отчета (на отдельном листе(-ах)): титульный лист, формулировка задания (вариант), ход выполнения работы, результаты выполнения работы, выводы.

ОСНОВНАЯ ЛИТЕРАТУРА

1. Карпова, Т.С. Базы данных: модели, разработка, реализация : учебное пособие / Т.С. Карпова. - 2-е изд., исправ. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 241 с. : ил. ; То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=429003

2. Давыдова, Е.М. Базы данных [Электронный ресурс] : учеб. пособие / Е.М. Давыдова, Н.А. Новгородова. — Электрон. дан. — Москва : ТУСУР, 2007. — 166 с. — Режим доступа: https://e.lanbook.com/book/11636. — Загл. с экрана.

3. Харрингтон, Д. Проектирование объектно ориентированных баз данных [Электронный ресурс] — Электрон. дан. — Москва : ДМК Пресс, 2007. — 272 с. — Режим доступа: https://e.lanbook.com/book/1231. — Загл. с экрана.

ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА

4. Голицина О.Л., Максимов Н.В., Попов И.И. Базы данных: учеб. пособие. – М.: Форум:Инфра-М, 2007.

5. Гагарин Ю.Е. Применение языка SQL в MS Access: учебно-методическое пособие. – М.: МГТУ им. Н.Э. Баумана, 2012.

Электронные ресурсы:

1. Научная электронная библиотека http://eLIBRARY.RU
2. Электронно-библиотечная система http://e.lanbook.com
3. Электронно-библиотечная система «Университетская библиотека онлайн» http://biblioclub.ru
4. Электронно-библиотечная система IPRBook http://www.iprbookshop.ru