СИСТЕМА
УПРАВЛЕНИЯ
БАЗАМИ
ДАННЫХ

ЛИНТЕР®

ЛИНТЕР БАСТИОН

ЛИНТЕР СТАНДАРТ

Модель защиты данных

НАУЧНО-ПРОИЗВОДСТВЕННОЕ ПРЕДПРИЯТИЕ

РЕЛЭКС

Товарные знаки

РЕЛЭКС™, ЛИНТЕР® являются товарными знаками, принадлежащими АО НПП «Реляционные экспертные системы» (далее по тексту – компания РЕЛЭКС). Прочие названия и обозначения продуктов в документе являются товарными знаками их производителей, продавцов или разработчиков.

Интеллектуальная собственность

О документе

Материал, содержащийся в данном документе, прошел доскональную проверку, но компания РЕЛЭКС не гарантирует, что документ не содержит ошибок и пропусков, поэтому оставляет за собой право в любое время вносить в документ исправления и изменения, пересматривать и обновлять содержащуюся в нем информацию.

Контактные данные

394006, Россия, г. Воронеж, ул. Бахметьева, 2Б.

Тел./факс: (473) 2-711-711, 2-778-333.

e-mail: market@relex.ru.

Техническая поддержка

С целью повышения качества программного продукта ЛИНТЕР и предоставляемых услуг в компании РЕЛЭКС действует автоматизированная система учёта и обработки пользовательских рекламаций. Обо всех обнаруженных недостатках и ошибках в программном продукте и/или документации на него просим сообщать нам в раздел Поддержка на сайте ЛИНТЕР.

Содержание

Предисловие

Назначение документа

Настоящий документ содержит описание модели защиты, задающей принцип разграничения доступа, описание правил разграничения доступа (ПРД) и их изменения, описание механизма управления доступом.

В описание модели защиты включены следующие положения:

- использование высокоуровневой спецификации части комплекса средств защиты (КСЗ), реализующего механизм управления доступом и его интерфейсов;
- верификация высокоуровневой спецификации на соответствие заданных принципов разграничения доступа;
- отображение высокоуровневой спецификации КСЗ последовательно в спецификации одного или нескольких нижних уровней, вплоть до реализации высокоуровневой спецификации КСЗ на языке программирования высокого уровня.

Документ предназначен для СУБД ЛИНТЕР СТАНДАРТ 6.0 сборка 20.1, далее по тексту СУБД ЛИНТЕР.

Для кого предназначен документ

Данный документ предназначен для использования администратором безопасности системы в части описания способов использования комплекса средств защиты (КСЗ) информации от несанкционированного доступа и администратором защиты в части описания модели защиты информации и гарантий проектирования.

Необходимые предварительные знания

Для понимания модели необходимо:

- знать основы реляционных баз данных;
- знать язык баз данных SQL.

Дополнительные документы

- СУБД ЛИНТЕР. Архитектура СУБД
- СУБД ЛИНТЕР. Справочник по SQL
- СУБД ЛИНТЕР. Администрирование комплекса средств защиты данных

Модель защиты

СУБД ЛИНТЕР представляет собой программное изделие, реализующее реляционный метод хранения данных и доступа к ним.

Архитектура СУБД ЛИНТЕР построена таким образом, что она позволяет выполнять параллельно несколько операций над данными различных пользователей. При хранении и обработке данных различных пользователей могут возникнуть ситуации, связанные с необходимостью ограничения доступа пользователей к данным друг друга. Для решения данной проблемы СУБД ЛИНТЕР предоставляет развитый аппарат защиты.

Модель доступа к данным

КСЗ СУБД ЛИНТЕР основываются на следующей модели доступа.

Любая реляционная СУБД хранит данные в виде таблиц. Таблица представляет собой обобщенное понятие, которое характеризуется некоторой структурой – структурой таблицы. Структура таблицы представляет собой список полей определенного типа. В таблицу могут быть занесены данные только со структурой, совпадающей со структурой таблицы.

Таблицы могут располагаться на различных внешних устройствах. В случае если база данных (БД) используется более чем одним пользователем или к системе, где хранятся данные, возможен доступ других пользователей, возникают вопросы конфиденциальности информации, санкционированности ее изменения или удаления.

Элементы защиты

Для решения вопросов организации доступа, в СУБД ЛИНТЕР рассматриваются следующие элементы:

Пользователь

Одно из основных понятий защиты. Под пользователем понимается некоторый идентификатор, обладающий некоторыми характеристиками. Все люди, которые должны получить доступ к данным БД, должны иметь подобный идентификатор.

Субъект контроля

Пользователь БД.

Администратор БД, системный пользователь

Стандартный идентификатор, автоматически создаваемый системой при создании новой БД.

Пароль пользователя

Характеристика пользователя, которая является конфиденциальной от других пользователей системы. Пользователь может изменять эту характеристику с целью понижения возможности для других пользователей получить доступ от его имени.

Пароль администратора безопасности

Характеристика администратора безопасности. Задается при создании БД и должна быть известна только администратору безопасности с целью предотвратить возможность других пользователей получить возможности администратора.

Идентификация пользователя

При открытии логической связи с СУБД пользователь обязан послать свое имя в качестве доказательства легальности его последующих действий. Данное имя должно быть известно БД, в противном случае, попытка установить логическую связь будет неуспешной.

Аутентификация пользователя

При открытии логической связи с СУБД ЛИНТЕР пользователь обязан послать свой пароль для подтверждения того факта, что канал открывает именно тот пользователь, который знает пароль, а не кто-то другой. В случае если пароль не совпадет с паролем, заданным легальным пользователем, запрос на подключение будет отвергнут.

Запрос к БД

Текст на специальном языке, который создает пользователь и передает СУБД ЛИНТЕР на обработку, содержащий указания, какие действия данный пользователь хочет проделать с БД.

Принудительное управление доступом

Механизм, реализуемый СУБД ЛИНТЕР, предназначенный для обеспечения высокой степени избирательности действия механизмов защиты данных.

Метка доступа

Совокупность нескольких параметров, проводящих в жизнь механизм принудительного управления доступом.

Параметры метки доступа

Набор нескольких числовых параметров и назначений, зависящих от типа помечаемого объекта.

Помеченные объекты

Объекты БД, которые имеют метку доступа СУБД ЛИНТЕР.

Группа доступа

Группа представляет собой уникальный идентификатор, соответствующий числовому значению в диапазоне от 1 до 250. Данные, принадлежащие одной группе, недоступны пользователям другой группы, если не указано доверие к группе. Группы доступа (одна или несколько) являются частью меток доступа.

Доверие к группе

Одна группа может доверить другой группе работу со своими данными.

Уровень доступа

Уровень доступа представляет собой идентификатор, соответствующий числовому значению в диапазоне от 1 до 10. Уровни доступа являются неотъемлемой частью меток доступа.

Метка пользователя

Включает в себя группу доступа, к которой принадлежит пользователь, уровни доступа пользователя и доверия пользователя.

Группа доступа пользователя

Группа, принадлежность к которой была указана при создании пользователя или в которую он был перемещен администратором безопасности.

Уровень доступа пользователя

Уровень, заданный при создании пользователя или назначенный ему позже, выше которого данный пользователь не может получать данные БД.

Уровень доверия пользователя

Уровень, заданный при создании пользователя или назначенный ему позже, ниже которого данный пользователь не может пометить информацию, заносимую им в БД.

Данные

Любая информация, которая хранится в БД, пересылается от СУБД к пользователю и от пользователя к СУБД. Данные могут иметь метку доступа.

Метка данных

Совокупность трех параметров – группы данных, уровня чтения данных и уровня записи данных. Данные могут не иметь метки только при передаче от пользователя в БД. При хранении в БД (при передаче пользователю) данные всегда имеют метку доступа.

Группа данных

Характеризует группу, от имени которой данные были записаны в БД или (при передаче от пользователя), от имени которой они заносятся в БД.

Уровень чтения данных

Указывает минимальный уровень доступа пользователя, который необходим для получения данных.

Уровень доступа данных

Указывает минимальный уровень пользователя, необходимый для изменения или удаления данной информации.

Атрибут

Атомарный объект, с которым работает БД. Представляет собой поле определенного типа, которое может содержать данные только соответствующего типа. Все атрибуты имеют метку доступа.

Метка доступа атрибута

Совокупность группы создателя атрибута и уровня чтения атрибута и уровня записи атрибута.

Уровень чтения атрибута

Ограничивает сверху уровень доступа пользователя при получении данных атрибута.

Уровень записи атрибута

Ограничивает снизу уровень доступа пользователя, необходимый для занесения данных в атрибут.

Структура таблицы

Совокупность атрибутов, представляющих собой структуру записи, которую можно вносить в таблицу.

Таблица

Поименованный структурированный объект, который может хранить данные. Все операции над данными БД требуют указания как минимум одной таблицы, с которой будет происходить работа. Таблица всегда имеет владельца. Таблица имеет метку доступа.

Метка доступа таблицы

Состоит из группы принадлежности таблицы, уровня чтения таблицы и уровня записи таблицы.

Группа принадлежности таблицы

Совпадает с группой владельца таблицы. При изменении группы владельца, автоматически сменится и принадлежность таблицы.

Уровень чтения таблицы

Уровень доступа, ограничивающий уровень доступа пользователя снизу для получения данных из таблицы. Автоматически данное условие необходимо и для любых других действий с данными таблицы.

Уровень записи таблицы

Уровень доступа, ограничивающий снизу уровень чтения данных, которые могут быть помещены в таблицу.

Имя таблицы

Имя соответствующего объекта.

Владелец таблицы

Пользователь, создавший данную таблицу.

Полное имя таблицы

Строка вида «имя пользователя.имя таблицы», которая однозначно идентифицирует конкретную таблицу в БД.

Представление

Является поименованным объектом, представляющим собой результат какого-либо запроса из таблиц или других представлений. Может использоваться в запросах вместо имени таблицы.

Полное имя представления

Строка вида «имя пользователя.имя представления», которая однозначно идентифицирует конкретное представление в БД. Не должно совпадать ни с одним из полных имен таблиц или других представлений.

Синоним

Идентификатор, эквивалентный полному имени таблицы или представления. Может использоваться вместо полного имени таблицы в запросах.

Право пользователя

Каждый пользователь для каждой таблицы может иметь определенные права. Получать информацию из нее, изменять, удалять или добавлять данные в таблицу. Существует право на изменение структуры таблицы.

Назначение права

Подача пользователем специального запроса, результатом которого является разрешение кому-либо некоторых действий с таблицами данного пользователя.

Роль

Совокупность прав, которые могут быть поименованы и, в совокупности, быть назначенными пользователю или отобраны у него.

Имя роли

Имя набора прав, которое может участвовать в запросах на назначение или отбор прав у пользователя, подразумевая совокупное назначение (или отбор) всех соответствующих роли прав.

Структура БД

Совокупность таблиц, синонимов, представлений, пользователей, прав пользователей, ролей.

Изменение структуры БД

Создание и удаление таблиц, представлений, синонимов, изменение структуры существующих таблиц, создание, удаление и модификация пользователей и ролей.

Категория пользователя

Характеристика пользователя, которая определяет, какие действия со структурой БД может производить данный пользователь.

Категория Connect

Не позволяет пользователю производить любые изменения структуры БД. Пользователь может изменять только собственный пароль.

Категория Resource

Позволяет пользователю изменять структуру БД в пределах, которые не затрагивают других пользователей. Пользователь может создавать и удалять таблицы, представления и синонимы, назначать права на свои таблицы другим пользователям, изменять структуру своих или чужих таблиц.

Категория DBA

Позволяет пользователю влиять на других пользователей. Создавать новых или удалять старых, изменять характеристики существующих пользователей.

Запись данных

Информационная последовательность, хранящаяся в таблице и совпадающая по структуре со структурой таблицы, всегда имеет метку доступа.

Метка доступа записи данных

Совокупность трех параметров – группы, уровня конфиденциальности и уровня ценности.

Группа записи данных

Группа доступа пользователя, который внес данную запись.

Уровень конфиденциальности (уровень чтения) записи данных

Классификационный уровень данных, указывающий уровень конфиденциальности информации данной записи.

Уровень ценности записи данных

Уровень данных, который указывает, насколько ценна данная запись. Используется при удалении или перезаписи данных.

Структура записи данных

Запись данных состоит из полей данных, структура которых совпадает со структурой таблицы, к которой относится данная запись. Поля данных могут иметь персональные метки безопасности. Поля данных, у которых нет персональных меток безопасности, считаются защищенными меткой доступа для записи.

Метка безопасности поля данных

Состоит из группы принадлежности поля данных и уровней конфиденциальности и ценности поля данных.

Уровень конфиденциальности поля данных

Классификационный уровень данных, указывающий уровень конфиденциальности, информации данного поля записи.

Уровень ценности поля данных

Уровень данных, который указывает, насколько ценно значение данного поля. Используется при изменении записи.

Сетевой адрес

Уникальный адрес в сети, позволяющий однозначно идентифицировать источник или приемник информации в сети.

Сетевое устройство

Устройство вне зависимости от физической реализации, имеющее уникальный идентификатор и соответствующий ему уникальный сетевой адрес. С устройства можно, используя тот или иной механизм, подавать запросы к СУБД. Сетевое устройство может иметь метку доступа.

Помеченное сетевое устройство

Сетевое устройство, имеющее сетевой адрес, известный БД, и обладающее уникальным именем внутри СУБД.

Непомеченное сетевое устройство

Сетевое устройство, сетевой адрес которого неизвестен БД.

Сетевой доступ

Доступ к ресурсам СУБД ЛИНТЕР с сетевого устройства. Доступ с помеченного устройства подчиняется правилам проверки мандатного доступа. Доступ с непомеченных устройств может быть запрещен администратором безопасности.

Параллельная работа

Возможность одновременной работы нескольких пользователей с различных сетевых устройств.

Архитектура параллельной работы

Внутренняя архитектура СУБД ЛИНТЕР, обеспечивающая преобразование длительных запросов пользователей в более мелкие и квантующая запросы пользователей на уровне более мелких запросов. Архитектура параллельной работы СУБД ЛИНТЕР не предусматривает исполнения программного кода пользователей и не предусматривает истинно параллельной работы. Все квантованные запросы выполняются последовательно. Архитектура СУБД также не предусматривает использование адресного пространства пользователя и не предоставляет пользователю доступа к собственному адресному пространству.

Адресное пространство

Пространство виртуальной памяти, выделенное операционной системой для выполнения отдельных процессов пользователей.

Адресное пространство СУБД ЛИНТЕР

Пространство виртуальной памяти, выделенное операционной системой для выполнения программного кода СУБД ЛИНТЕР. Изоляция данного адресного пространства от адресных пространств пользователей системы обеспечивается операционной системой.

Метка сетевого устройства

Содержит маску доступа для групп пользователей. Уровень защищенности устройства и уровень доверия устройства.

Маска доступа сетевого устройства

Битовая маска, определяющая, может ли пользователь соответствующей группы устанавливать логическое соединение с БД с данного устройства.

Уровень защищенности сетевого устройства

Ограничивает сверху уровень доступа пользователя, устанавливающего соединение с СУБД.

Уровень доверия сетевого устройства

Ограничивает снизу уровень доверия пользователя, устанавливающего соединение с СУБД.

ЛИНТЕР-устройство

Уникальный четырехбуквенный идентификатор, который может быть указан в запросах к БД на создание таблиц. ЛИНТЕР-устройство соответствует какому-либо физическому устройству ввода-вывода.

Физическое устройство ввода-вывода

Любое устройство постоянного хранения данных вне зависимости от расположения или внутренней структуры, на котором могут быть созданы файлы БД (таблицы или временные файлы). Устройства могут иметь метку доступа.

Непомеченное физическое устройство ввода-вывода

ЛИНТЕР-устройство, неизвестное СУБД. Соответствие между таким устройством и физическим расположением файлов БД производится на уровне операционной системы. Доступ с непомеченных устройств может быть запрещен или разрешен администратором безопасности.

Помеченное физическое устройство ввода-вывода

ЛИНТЕР-устройство, известное СУБД. Такие устройства всегда имеют метку доступа.

Метка доступа физического устройства

Состоит из маски доступа групп, уровня конфиденциальности устройства и уровня целостности устройства.

Маска доступа групп к физическому устройству

Определяет группы пользователей, члены которых имеют возможность создавать на данном устройстве объекты БД (таблицы).

Уровень конфиденциальности физического устройства

Уровень конфиденциальности физического устройства представляет собой минимальный уровень доступа пользователя, необходимый для создания этим пользователем объектов (таблиц и временных файлов) на данном физическом устройстве.

Уровень целостности физического устройства

Уровень целостности физического устройства представляет собой максимальный уровень доверия пользователя необходимый, для удаления таблицы, расположенной на данном физическом устройстве.

Использование ресурсов памяти системой

СУБД ЛИНТЕР использует оперативную память внутри своего адресного пространства и внешние устройства постоянного хранения информации. Оперативная память используется для повышения производительности системы. Внешняя память используется для хранения данных пользователей.

Использование оперативной памяти

Использование оперативной памяти системой ЛИНТЕР находится под контролем подсистемы безопасности. Все использованное оперативное пространство учитывается и переиспользуется. СУБД в процессе работы не выполняет операций освобождения оперативной памяти. Передача пространства оперативной памяти под контроль операционной системы происходит только при закрытии системы. При этом КСЗ производит все необходимые процедуры очистки.

Использование внешней памяти

СУБД ЛИНТЕР хранит данные пользователей в файлах БД, располагающихся на внешних устройствах. В процессе работы система может запросить дополнительную память. Освобождение внешней памяти происходит только при уничтожении объектов БД (таблиц). Все запросы на размещение или освобождение внешней памяти контролируются КСЗ, что позволяет провести процедуры очистки памяти перед передачей ее под контроль операционной системы.

Регистрация событий

СУБД ЛИНТЕР позволяет регистрировать различные события, происходящие в БД, которые могут быть связаны с попытками нарушения доступа, ошибками проектирования системы или при контроле за использованием данных. Данная информация может периодически анализироваться на предмет выявления нарушений или несоответствий в работе КСЗ.

Ядро СУБД ЛИНТЕР

Ядро представляет собой программный модуль, выполняющий все действия по работе с данными и СУБД.

Целостность КСЗ

КСЗ СУБД ЛИНТЕР являются неотъемлемой частью ядра СУБД и его целостность эквивалентна целостности ядра в целом.

Схема защиты

Модель защиты СУБД ЛИНТЕР, опираясь на вышеперечисленные элементы, обеспечивает полнофункциональную многоуровневую схему защиты данных на всех этапах обработки и хранения данных в системе:

- СУБД ЛИНТЕР работает только с идентифицированными пользователями;
- все данные, хранящиеся в БД, имеют владельца и идентификационные метки;
- всем пользователям БД ставится в соответствие список прав доступа;
- пользователь может выполнить операцию только в случае, если операция будет разрешена всеми уровнями защиты одновременно;
- администраторы не имеют непосредственно доступа к данным других пользователей. Их отличия от остальных заключаются только в возможности управлять другими пользователями;

- администратор БД не имеет непосредственно доступа к данным. Его функции по сравнению с остальными субъектами расширяются только в сторону управления системой безопасности в целом;
- все передаваемые под контроль операционной системы ресурсы не несут информации, содержавшейся в них во время работы СУБД ЛИНТЕР;
- администратор БД может контролировать работу пользователей в многопользовательской системе на основе развитых средств регистрации событий;
- для многопользовательской сетевой работы может применяться принудительное управление доступом;
- пользователь может получить метки доступа запрашиваемых данных с целью дальнейшего контроля за дальнейшим использованием информации;
- все действия по каналам логической связи с БД надежно связаны с пользователем, который открыл данный канал. После закрытия логической связи невозможно пользоваться данным каналом;
- СУБД ЛИНТЕР использует принцип минимальных привилегий в случае, если привилегии не указываются явно.

Виды нарушений

Данная модель препятствует следующим видам нарушений:

- действия незарегистрированного пользователя;
- действия нарушителя от имени легального пользователя;
- получение данных без разрешения владельца;
- получение информации с высоким уровнем конфиденциальности пользователем с низким уровнем конфиденциальности;
- понижению уровня конфиденциальности данных;
- извлечению информации из пространств памяти, переданной под контроль операционной системы;
- размещению данных на выделенных устройствах;
- подключению пользователей со слабо защищенных устройств сети;
- неконтролируемому распространению конфиденциальной информации после выдачи ее из БД;
- падению надежности системы при сбоях в работе оборудования;
- присвоению пользователем себе новых прав;
- нарушениям целостности КСЗ;
- нарушениям, возникающим при ошибках администрирования БД.

Объекты контроля КСЗ

Объектами контроля в СУБД ЛИНТЕР являются *таблицы* (базовые или виртуальные), представления, а также более мелкие элементы данных: *столбцы* и *строки* таблиц и даже поля строк.

Таблицы БД и *представления* – именованные объекты. Они имеют *владельца* или *создателя*.

Их объединяет еще и то, что все они для конечного пользователя представляются как таблицы, т.е. как нечто именованное, содержащее информацию в виде множества строк (записей) одинаковой структуры. Строки таблиц разбиты на поля именованными столбцами. Все поля соответствующие одному столбцу имеют один и тот же тип и длину данных.

Кроме того, в ЛИНТЕР таблицы (представления) разных пользователей могут иметь одинаковые имена, так что при обращении к таблице другого пользователя необходимо указывать полное имя таблицы, состоящее **из двух имен**: пользователя и таблицы.

Это то общее, что имеют все таблицы и представления с точки зрения КСЗ.

На самом деле информация реально хранится только в базовых таблицах.

Виртуальная таблица (в ЛИНТЕР это только таблица каналов) – это только табличное представление текущего состояния одной из очередей ядра системы. Информацию, которую ”содержит” виртуальная таблица, система черпает из той оперативной памяти, которую занимает.

Представление (View) – это только SQL-ссылка на базовые или виртуальные таблицы. Представление – всего лишь SQL-запрос, выборка из базовых/виртуальных таблиц. Эта выборка образуется каждый раз при обращении к представлению.

Более мелкие элементы данных, такие как столбцы и строки таблиц, поля строк (значения) так же являются объектом защиты.

Все объекты защиты перечислены в таблице 1.

Таблица 1. Объекты защиты СУБД ЛИНТЕР

Объект защиты	Логическая защита	Физическая защита
Базовая таблица	+	+
Представление	+	
Столбец таблицы		+
Строка таблицы		+
Поле		+

Логическая и физическая защита в главном различаются между собой тем, что логическую защиту объектов (таблиц, представлений) можно модифицировать (изменять возможности доступа к этим объектам), а физическая защита (метки доступа) жестко закреплена за данными, не модифицируема и удаляется только вместе с данными.

Логическая защита

Логическая защита в ЛИНТЕР представляет собой набор прав субъектов или ролей по отношению к защищаемому объекту. Кроме того, к логической защите можно отнести и владение таблицей (представлением).

Набор прав (логическую защиту) владелец таблицы может изменять (расширять, отнимать, ограничивать доступ).

Отметим еще и тот факт, что данные о логической защите находятся в системных таблицах БД и отделены от защищаемых объектов (от таблиц или представлений).

Физическая защита

Физическая защита ЛИНТЕР характеризует, главным образом, данные (их принадлежность, важность, представительность и пр.). Это в основном метки безопасности, описывающие группу принадлежности и уровни конфиденциальности и ценности данных объекта (таблицы, столбца, строки или поля).

Метки безопасности (физическая защита) неизменны на всем протяжении существования объекта защиты (умирают только вместе с ним) и территориально (на диске) располагаются вместе с защищаемыми данными.

Кроме атрибута собственности (логическая защита), разбивающего данные (таблицы) на собственные (принадлежащие данному субъекту) и чужие, физическая защита разбивает данные более тонко.

Во-первых, все перечисленные объекты (независимо от их иерархии в БД) разбиваются на *группы принадлежности*. Объект может принадлежать только одной из групп (это может быть, например, разбиение по отделам организации).

Группы принадлежности напрямую связаны с *группами субъектов* (см. ниже). Субъект вправе **видеть** только данные своей группы, если между группами субъектов не установлены отношения доверия.

Во-вторых, все объекты выстроены в иерархию по *уровням конфиденциальности* и по *уровням ценности* или важности.

Уровень конфиденциальности разбивает объекты по доступности на *чтение* (и даже на *видение*). Пользователь с более низким уровнем доступа не будет знать даже о существовании объектов с более высоким уровнем конфиденциальности.

Уровень ценности, напротив, разбивает данные (объекты) по важности, ограничивая возможность их удаления и модификации.

Субъекты контроля КСЗ

Каждый пользователь, соединяющийся с СУБД ЛИНТЕР, должен пройти процедуры *идентификации* и *аутентификации*.

При идентификации пользователь указывает свое имя. Следующим этапом проверяется подлинность идентификации. Это, так называемая, аутентификация, т.е. проверка подлинности имени с помощью пароля пользователя.

Соединение с системой неидентифицированных пользователей и пользователей, подлинность идентификации которых при аутентификации не подтвердилась, **исключается**.

В процессе сеанса работы пользователя (от удачного прохождения идентификации и аутентификации до отсоединения от системы) все его действия надежно связываются с результатом идентификации.

Отсоединение пользователя может быть как *нормальным* (исходить от этого же пользователя, по тому же каналу), так и *насильственным* (исходящим от пользователя-администратора, например, в случае удаления пользователя или аварийном удалении канала связи). В случае насильственного отсоединения пользователь будет проинформирован об этом, все его действия аннулированы до последней фиксации изменений, произведенных им в таблицах БД.

В любом случае на время сеанса работы идентифицированный пользователь будет для КСЗ СУБД ЛИНТЕР субъектом контроля.

Следуя технологии открытых систем, субъект контроля может обращаться посредством СУБД ЛИНТЕР к БД **только из программ** (поставляемых в дистрибутиве или подготовленных им самим) и только с помощью штатных средств системы.

Все субъекты контроля системы хранятся в таблице полномочий системы и разделены для системы на три категории: `Connect`, `Resource` и `DBA`.

`Connect`

Категория дает право на подсоединение к системе и подачу запросов к доступным ему данным.

`Resource`

Категории доступны все возможности Connect-категории плюс возможности изменения структуры БД путем построения своих объектов контроля (таблиц, представлений, синонимов).

`DBA`

Категория (категория администраторов БД) включает возможности обеих предыдущих категорий, а также возможность вводить (удалять) в систему (из системы) субъекты контроля или изменять уровень их полномочий (категорию).

Итак, по возможностям эти категории можно расположить в следующем порядке:

категория `DBA` включает в себя привилегии категории `RESOURCE` и категория `RESOURCE` включает в себя привилегии категории `CONNECT`, т.е. категории упорядочены.

Это первое и достаточно важное деление субъектов контроля, принятое в СУБД ЛИНТЕР.

Можно более тонко разделить (по доступу) субъекты контроля и с помощью ролей. Категории доступа и роли позволяют в полном объеме реализовать *дискреционный принцип контроля*.

Мандатный принцип контроля, реализованный в СУБД ЛИНТЕР, предлагает дополнительное деление пользователей на:

- *группы доступа* (не путать с ролями);
- иерархию по *уровням доступа* (десять уровней с номерами от 1 до 10);
- иерархию по *уровням доверия* (десять уровней с номерами от 1 до 10).

Группа доступа субъекта

Группа доступа субъекта (допущенного до работы пользователя) указывает на его принадлежность к группе субъектов по доступу (это может быть, например, принадлежность к отделу организации, к группе людей, объединенных одними функциями и пр.). Группа назначается субъекту *администратором безопасности*, последний может так же переместить субъекта из одной группы в другую.

Всего (в СУБД ЛИНТЕР) может быть 250 таких групп с номерами от 1 до 250.

Группы доступа пользователей напрямую связаны с группами данных. Данным (объектам) присваивается группа доступа пользователя (субъекта), который их внес. Данные, принадлежащие одной группе, недоступны пользователям другой группы. Однако одна группа может *доверить* другой группе работу со своими данными.

Уровень доступа субъекта

Уровень доступа субъекта задается при его создании (администратором безопасности) или изменяется позже. Он определяет: какие из данных (по уровню конфиденциальности) доступны субъекту, а какие нет. Все данные с уровнем конфиденциальности более высоким, чем уровень доступа субъекта, последнему недоступны и он даже не подозревает об их существовании.

Уровень доверия субъекта

Уровень доверия субъекта (или уровень доверия субъекта на понижение уровня конфиденциальности) назначается ему администратором безопасности.

Этот уровень определяет *важность* (т.е. *конфиденциальность*) данных, вносимых субъектом. Система ЛИНТЕР не позволит ему внести информацию (объект) с низким (ниже по значению, чем уровень доверия субъекта) уровнем конфиденциальности.

Этот уровень призван защитить важных персон от случайного (или намеренного) присвоения секретным данным низкого уровня конфиденциальности. Любая вносимая (изменяемая) этим субъектом информация уже будет иметь минимальный «гриф» или уровень конфиденциальности. Его информация может быть только более (не менее) защищенной.

Правила разграничения доступа

В СУБД ЛИНТЕР реализуемы *дискреционный* и *мандатный* принципы контроля доступа.

При этом действуют **два глобальных правила**:

1) доступ к объектам, имеющим дискреционную и мандатную защиту, должен быть санкционирован ими обоими. Если хотя бы одна из них (дискреционная или мандатная) отвергает доступ, то запрос на доступ будет отвергнут по обоим правилам (принцип эквивалентности);
2) при отсутствии у субъекта доступа к какому-либо объекту по одному из принципов (если в отношении субъекта и объекта действуют оба принципа), он не сможет ни управлять доступом к этому объекту, ни получить доступ к этому объекту. В этом отношении среди пользователей выделяется только администратор БД, который может изменять метки доступа пользователей (но не данных и их меток). Метки доступа позволяют осуществлять многоуровневую защиту.

Дискреционный принцип контроля доступа

Для каждой пары субъект-объект можно задать явное и недвусмысленное перечисление возможных действий субъекта (таблица 2).

Таблица 2. Допустимые действия субъекта

Действие	Описание
SELECT	*Чтение* данных объекта
INSERT	*Добавление* новых данных в объект
DELETE	*Удаление* некоторых/всех данных объекта
UPDATE	*Изменение* данных объекта
ALTER	*Изменение физической/логической структуры* базовой таблицы (изменение размеров и числа файлов таблицы, введение дополнительного столбца и т.п.)
INDEX	*Создание/удаление индексов* на столбцы базовой таблицы
REFERENCE	*Запрещение/разрешение создавать ссылки* на столбец (столбцы) таблицы
BACKUP	*Запрещение/разрешение архивирования* объектов БД
ALL	Все возможные действия, т.е. все предыдущие действия вместе взятые

Разрешение тех или иных действий над объектом для какого-то субъекта – прерогатива владельца этого объекта. Только субъект-владелец объекта может (рисунок 1) предоставить некоторые/все права на этот объект другим (и даже всем) субъектам.

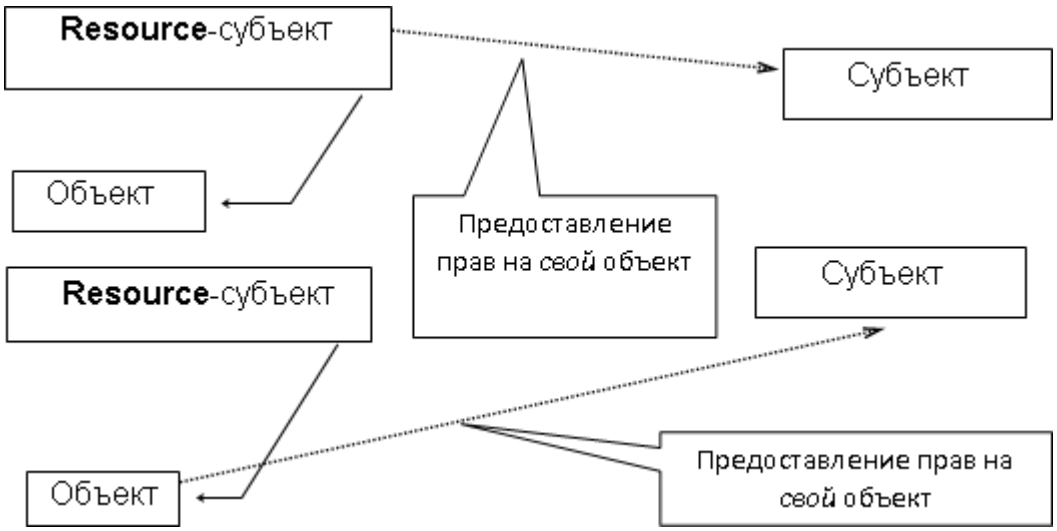

Рисунок 1. Схема прямой передачи доступа в СУБД ЛИНТЕР

В серьезных БД, где много (тысячи) субъектов и объектов, очень сложно определять возможности (или права) каждого субъекта при работе с каждым объектом. Для упрощения этой процедуры в ЛИНТЕР реализован аппарат *ролей*.

Роль – это **именованный набор прав на множество объектов** БД. Введенные роли разбивают всех пользователей БД на группы. Те, кому присвоена (назначена) определенная роль, образуют естественную группу.

При работе с привилегиями роль рассматривается сначала как пользователь или субъект. Владелец того или иного объекта БД назначает тот или иной вид доступа для роли как для простого пользователя. После того, как роль вобрала в себя все требуемые возможности, она может быть присвоена пользователю или группе пользователей.

Итак, инициатором передачи/ограничения прав могут быть только владельцы информации, только владельцы объектов. В этом и проявляется **принцип ограничения распространения прав** на доступ.

Роль может быть присвоена не только пользователю, но и другой роли, образуя иерархию. Имеющуюся у пользователя роль (роли) можно отнять.

С помощью аппарата ролей можно строить не только иерархические взаимоотношения пользователей (старший – подчиненный), но и вообще произвольную структуру доступа.

Возникают естественные вопросы о том:

1) кто может создать роль?
2) кто имеет право назначить или отнять роль?

Рисунок 2. Схема построения ролей в СУБД ЛИНТЕР

Из рисунка 2 следует, что:

1) создать роль (и, значит стать ее владельцем) может только субъект DBA-категории;
2) назначить/отнять роль может только ее владелец/создатель.

Отсюда следует еще и то, что владельцы объектов могут всегда в процессе работы отменить некоторые права, ранее данные ролям, что повлечет за собой изменение иерархии прав на более высоких уровнях.

Аналогично, субъекты-администраторы могут отнять некоторые свои роли, ранее назначенные более верхним ролям, и это также повлечет за собой изменение иерархии прав.

Итак, в СУБД ЛИНТЕР присутствуют две схемы передачи прав: напрямую и косвенно, через аппарат ролей. Причем обе эти схемы можно использовать в совокупности.

Примечания

1. Для доступа к объекту БД с требуемой привилегией данная привилегия должна быть назначена пользователю либо с помощью механизма назначения привилегий пользователям, либо с помощью механизма назначения привилегий роли, при назначении этой роли пользователю, либо при назначении привилегии доступа к объекту `PUBLIC` (под `PUBLIC` имеется в виду команда вида "grant <привилегия> on <объект> to PUBLIC").
   То есть, для пользователя действует совокупность (объединение) всех привилегий дискреционного доступа: полученных напрямую, через роли, через `PUBLIC` и привилегии, которые он имеет, как владелец объектов.

2. Доступ ко всем объектам, поддерживаемым системой возможен только через механизм контроля доступа и только для тех субъектов, которые зарегистрированы в системе.

Мандатный принцип контроля доступа

Примечание
Поддерживается только в СУБД ЛИНТЕР БАСТИОН.

Мандатный принцип контроля доступа основан на разграничении доступа субъектов к объектам с помощью назначения *метки конфиденциальности* (*метки доступа*) для информации, содержащейся в объектах, и выдаче официальных разрешений (допуска) субъектам на обращение к информации такого уровня конфиденциальности.

Метка доступа хранится вместе с объектом защиты и играет важную роль при контроле доступа субъекта к информации помеченного объекта.

1) Для пользователя (субъекта) вводятся следующие уровни доступа:
   - RAL-уровень доступа. Пользователь имеет доступ к информации, RAL-уровень которой не выше его собственного уровня доступа;
   - WAL-уровень доверия на понижение уровня конфиденциальности. Пользователь не может вносить информацию с уровнем доступа (RAL) более низким, чем данный WAL-уровень пользователя. Т.е. пользователь не может сделать доступную ему информацию менее конфиденциальной, чем указано в данном параметре.

   WAL-уровень доверия пользователя введен для того, чтобы ограничить возможности пользователя, имеющего высокий уровень доступа (RAL), вносить какую-либо информацию с более низким уровнем секретности. Эта возможность и определяется уровнем доверия, показывающим насколько на самом деле можно доверять пользователю, имеющему доступ к секретным данным, в том, что он не станет их рассекречивать.

   В предельном случае WAL-уровень пользователя может быть равен или даже выше его RAL-уровня. Последнее может иметь место для пользователей (например, *аналитиков*), которым разрешено видеть и анализировать информацию, но запрещено ее записывать (или, по крайней мере, разрешено записывать только с более высоким уровнем секретности).
2) Для информации (данных) вводятся следующие уровни доступа:
   - RAL-уровень чтения. Информация может быть прочитана пользователем, RAL-уровень которого не ниже RAL-уровня информации (т.е. пользователем, имеющим необходимый уровень доступа);
   - WAL-уровень ценности или уровень доступа на запись (модификацию, удаление). Информация может быть модифицирована (удалена) пользователем, RAL-уровень которого не ниже WAL-уровня модифицируемой информации.
3) Для таблиц и их столбцов вводятся следующие уровни доступа:
   - RAL-уровень чтения. Указывает минимальный уровень доступа пользователя (RAL), который необходим для получения любых данных из таблицы (столбца), т.е. RAL-уровень для таблиц и столбцов по смыслу идентичен RAL-уровню данных;
   - WAL-уровень записи. Ограничивает снизу уровень чтения данных (RAL), которые могут быть помещены в таблицу.

При мандатном контроле доступа пользователь не ограничивается в повышении уровня чтения данных. При желании он может заносить в БД информацию, RAL-уровень которой выше его собственного RAL-уровня. В частности, к таким пользователям можно отнести категорию *информаторов*, которые имеют самый низкий RAL-уровень. Они не могут читать никакой секретной информации, но могут ее записывать в БД. Чтобы дать им возможность записывать в БД информацию с более высоким уровнем секретности, можно поднять значение их WAL-уровня.

Уровни доступа субъекта и объекта БД сравниваются СУБД ЛИНТЕР при попытке получения доступа субъекта к объекту.

Метка доступа наследуется от метки доступа субъекта при создании им объекта. В случае изменения данных метка доступа не меняется. Это относится только к значениям по умолчанию. При явном задании значения оно может указываться вместе с меткой доступа.

Многоуровневая защита

Многоуровневая защита с помощью меток доступа обеспечивает разграничение доступа субъектов с различными правами доступа к объектам различных уровней конфиденциальности.

Уровни (доступа и ценности), входящие в метку доступа защищаемых объектов, как раз и призваны обеспечить подобную многоуровневую защиту.

При этом не только объекты разбиты на уровни, но и субъекты.

Рекомендации по созданию/изменению ПРД

Ниже приведены примеры по созданию и/или изменению прав разграничения доступа. В каждом примере конфигурации КСЗ БД существует выделенный пользователь – администратор БД. Он создается при генерации БД с помощью утилиты `gendb`. Имя первоначального администратора всегда SYSTEM.

Если в процессе создания БД имя и пароль администратора явно не указываются, например:

```
GENDB> CREATE DATABASE "TEST";
```

то по умолчанию администратор получает имя SYSTEM и пароль MANAGER8.

Возможно явное задание пароля администратора без указания его имени:

```
GENDB> CREATE DATABASE "TEST" PASSWORD "FIRST";
```

Администратор получает имя SYSTEM (по умолчанию) и пароль FIRST.

Справочная БД одного администратора

Все субъекты и все объекты БД создаются одним администратором.

Все пользователи получают категорию `Connect`. На все объекты БД им назначается привилегия `Select`.

Для такого конфигурирования администратор создает роль:

```
CREATE ROLE Select_Role;
```

Затем этой роли присваивается привилегия `Select` на все таблицы БД:

```
 GRANT SELECT ON Table_Name_1 TO Select_Role;
 GRANT SELECT ON Table_Name_2 TO Select_Role;
 GRANT SELECT ON Table_Name_3 TO Select_Role;
```

`Table_Name_N` – имена таблиц БД. Администратор является владельцем этих таблиц.

Теперь, после создания нового пользователя (например, `User1`), ему назначается роль `Select_Role`:

```
GRANT ROLE Select_Role TO User1;
```

В этом случае новый пользователь получает привилегию `Select` на все объекты БД.

В этом примере модификацию таблиц БД, изменение их структуры, создание индексов, создание новых таблиц может производить только администратор, являющийся владельцем всех таблиц.

Администратор и Connect-пользователи

Все субъекты и все объекты БД (как и в предыдущем разделе) создаются одним администратором.

Все остальные пользователи имеют Connect-категорию, но различные привилегии, образуя две группы:

1) пользователи, имеющие привилегии `Select+Insert+Delete+Update` на все таблицы;
2) пользователи, имеющие только привилегию `Select` на все таблицы.

Для такого конфигурирования администратор создает 2 роли: `SIDU_Role` и `S_Role`.

```
CREATE ROLE S_Role;
CREATE ROLE SIDU_Role;
```

Затем этим ролям присваиваются соответствующие привилегии на все таблицы БД:

```
GRANT SELECT ON Table_Name_1 TO S_Role;
GRANT SELECT ON Table_Name_2 TO S_Role;
GRANT SELECT ON Table_Name_3 TO S_Role;
GRANT SELECT,INSERT,UPDATE,DELETE ON Table_Name_1 TO SIDU_Role;
GRANT SELECT,INSERT,UPDATE,DELETE ON Table_Name_2 TO SIDU_Role;
GRANT SELECT,INSERT,UPDATE,DELETE ON Table_Name_3 TO SIDU_Role;
```

Теперь пользователю `User1`, вводимому в систему, можно назначить одну из двух ролей, т.е. отправить его в одну из обозначенных групп.

Для этого нужно выполнить один из запросов

```
GRANT ROLE S_Role TO User1;
```

для перевода в первую группу и

```
GRANT ROLE SIDU_Role TO User1;
```

для перевода во вторую.

И в этом примере владельцем всех объектов является администратор. Только он может создавать новые таблицы, изменять структуру старых. Модификацию таблиц уже могут производить специально назначенные пользователи (из второй группы).

Администратор и два независимых отдела

Пусть БД принадлежит организации, имеющей в своем составе 2 и более отделов, каждый из которых имеет свои таблицы, доступ к которым должен быть закрыт для сотрудников других отделов.

Сотрудники каждого отдела имеют привилегию чтения из таблиц своего отдела.

Для каждого отдела создается роль, которой назначается привилегия `Select` только на таблицы, с которыми необходимо работать в данном отделе.

```
CREATE ROLE Select_Sect1;
GRANT SELECT ON Table1_Sect1 TO Select_Sect1;
GRANT SELECT ON Table2_Sect1 TO Select_Sect1;
GRANT SELECT ON Table3_Sect1 TO Select_Sect1;
CREATE ROLE Select_Sect2;
GRANT SELECT ON Table1_Sect2 TO Select_Sect2;
GRANT SELECT ON Table2_Sect2 TO Select_Sect2;
GRANT SELECT ON Table3_Sect2 TO Select_Sect2;
```

Затем эти роли назначается пользователям.

Если пользователь `User_Name` принят в первый отдел, то нужно выполнить запрос

```
GRANT ROLE Select_Sect1 TO User_Name;
```

Если во второй, то запрос

```
GRANT ROLE Select_Sect2 TO User_Name;
```

В этом примере модификацию таблиц БД, изменение их структуры, создание индексов, создание новых таблиц может производить только администратор БД, являющийся владельцем всех таблиц.

Два неравноправных отдела

Рассмотрим БД организации, имеющей в своем составе 2 отдела, каждый из которых имеет свои таблицы. Сотрудники каждого из этих отделов имеют полный доступ к таблицам своего отдела. Кроме того, сотрудники отдела №1 имеют право читать таблицы отдела №2.

Для каждого отдела создается роль, которой назначается привилегия `All` (полный доступ) на таблицы, с которыми необходимо работать в данном отделе.

```
CREATE ROLE Sect_N1;
GRANT ALL ON Table1_Sect_N1 TO Sect_N1;
GRANT ALL ON Table2_Sect_N1 TO Sect_N1;
GRANT ALL ON Table3_Sect_N1 TO Sect_N1;
CREATE ROLE Sect_N2;
GRANT ALL ON Table1_Sect_N2 TO Sect_N2;
GRANT ALL ON Table2_Sect_N2 TO Sect_N2;
GRANT ALL ON Table3_Sect_N2 TO Sect_N2;
```

Затем для роли отдела №1 добавляется возможность чтения таблиц отдела №2.

```
GRANT SELECT ON Table1_Sect_N2 TO Sect_N1;
GRANT SELECT ON Table2_Sect_N2 TO Sect_N1;
GRANT SELECT ON Table3_Sect_N2 TO Sect_N1;
```


Далее каждому пользователю назначается роль в зависимости от отдела, в котором он работает.

Владельцем всех таблиц является администратор. Только он может создавать новые таблицы и вводить новых пользователей.

Информаторы, верификаторы и аналитики

Часто используемая у нас в стране и за рубежом система разграничения полномочий при сборе конфиденциальной информации, когда *информаторы*, т.е. те, кто собирают и заносят информацию, могут только вносить новую информацию, но не имеют право читать ее.

Информаторам часто запрещается видеть цельную картину, это удел так называемых *аналитиков*, в функции которых входит обработка и анализ собранной информации.

Однако между информаторами и аналитиками должно стоять еще одно звено – это пользователи с функциями проверки, т.е. *верификаторы*. Верификаторы должны производить, во-первых, проверку введенных информаторами фактов, во-вторых, первичную обработку (например, обезличивание фактов).

Следовательно, верификаторам должны быть разрешены выборка из таблиц информаторов и запись (с возможной модификацией) в таблицы аналитиков.

Обычно, аналитик результаты анализа собирает в **своих** таблицах, доступ к которым имеет только он и никто больше (другие аналитики и даже администратор).

В этом примере роли нужно построить так:

- роль информаторов (`Inf_Role`):

```
CREATE ROLE Inf_Role;
GRANT INSERT ON Inf_1 TO Inf_Role;
GRANT INSERT ON Inf_2 TO Inf_Role;
GRANT INSERT ON Inf_N TO Inf_Role;
```

- роль верификаторов (`Ver_Role`):

```
CREATE ROLE Ver_Role;
GRANT SELECT ON Inf_1 TO Ver_Role;
GRANT SELECT ON Inf_2 TO Ver_Role;
GRANT SELECT ON Inf_N TO Ver_Role;
GRANT SELECT,INSERT,UPDATE ON Ver_1 TO Ver_Role;
GRANT SELECT,INSERT,UPDATE ON Ver_2 TO Ver_Role;
GRANT SELECT,INSERT,UPDATE ON Ver_M TO Ver_Role;
```

- роль аналитиков (`Anal_Role`):

```
CREATE ROLE Anal_Role;
```

```
GRANT SELECT ON Ver_1 TO Anal_Role;
GRANT SELECT ON Ver_2 TO Anal_Role;
....
GRANT SELECT ON Ver_M TO Anal_Role;
```

При этом создание нового пользователя выглядит так:

- создание информатора:

```
GRANT CONNECT TO User_Name IDENTIFIED BY 'User_Password';
GRANT ROLE Inf_Role TO User_Name;
```

- создание верификатора:

```
GRANT CONNECT TO User_Name IDENTIFIED BY 'User_Password';
GRANT ROLE Ver_Role TO User_Name;
```

- создание аналитика:

```
GRANT RESOURCE TO User_Name IDENTIFIED BY 'User_Password';
GRANT ROLE Anal_Role TO User_Name;
```

Обращаем внимание, что владельцем всех таблиц, кроме аналитических, является администратор. Таблицы с результатами анализа данных скрыты от всех, кроме их владельцев-аналитиков.

Разграничение доступа в одной таблице

Когда речь идет о продажах, то возникает ситуация, когда даже в **одной** таблице различные пользователи должны ”видеть” только разрешенные им данные. Причем другие таблицы (например, таблица курса валют) доступны всем (публичные таблицы).

Примером может служить следующая (несколько упрощенная) схема БД:

- таблица (`Weapon`) – общий справочник вооружений;
- таблица характеристик оружия (`Descriptions`);
- таблица курса валют (`Currency`).

Во-первых, со второй таблицей будет работать группа узких специалистов, которым вовсе незачем ”видеть” то, в чем они не разбираются.

Для этого владелец таблицы `Descriptions` должен сделать из нее несколько **представлений** по числу специалистов (или групп специалистов), например, так:

```
CREATE VIEW "Легкие танки" AS
SELECT * FROM Descriptions WHERE Type ='Легкий танк';
```



```
CREATE VIEW "Тяжелые танки" AS
SELECT * FROM Descriptions WHERE Type ='Тяжелый танк';

CREATE VIEW "Истребители" AS
SELECT * FROM Descriptions WHERE Type ='Истребитель';
 …
```

Кроме того, в группе технических и маркетинговых специалистов, представляющих, скажем, самолеты-истребители, также возможно разделение по доступу. Не все столбцы (читай не все характеристики) представления `Истребители` должны быть доступны всем субъектам группы.

Например, такие характеристики, как институт-разработчик или завод-изготовитель можно выбирать только в исключительных случаях, когда необходима связь для получения более тонкой (полной) технической или продажной информации.

Это также можно сделать с помощью представлений. В предыдущих View таблица характеристик оружия разбита на части по строкам. В каждую из частей входят строки, относящиеся только к какому-то одному типу вооружений. Однако таблицу вооружений можно разбить на части и по столбцам, а также и по строкам, и по столбцам. К каждому представлению, отражающему ту или иную часть таблицы `Descriptions`, можно обычным образом (как для базовой таблицы) установить правила разграничения доступа.

Несколько администраторов

В БД может быть создано несколько пользователей с правами администратора БД (`DBA`). Для создания такого пользователя необходимо подать команду:

```
GRANT DBA TO User_Name IDENTIFIED BY 'User_Password';
```

`User_Password` – пароль вновь созданного пользователя с именем `User_Name`.

Каждый DBA-пользователь обладает всеми правами, доступными `Resource`, `Connect` и может создавать новых пользователей.

Здесь возможны любые комбинации с вышеизложенными примерами.

Для обеспечения надлежащего уровня безопасности БД не рекомендуется создавать больше 6 пользователей с уровнем прав `DBA`.