Практическое занятие 2. Назначение ролей сервера и базы данных

1. В чем разница между логином и пользователем БД?

Логин (Login):

-Логин — это объект, созданный на уровне серверной части базы данных (например, на уровне SQL Server или другого сервера БД).

-Используется для аутентификации, то есть проверки подлинности подключающегося субъекта (клиента, приложения, пользователя).

-Логину назначаются пароли или сертификаты для аутентификации.

-Логин контролирует доступ к самому серверу, а не к конкретной базе данных.

Пользователь (User):

-Пользователь создается на уровне отдельной базы данных.

-Связан с логином, но действует только в пределах той базы данных, к которой он прикреплен.

-Пользователю предоставляются права доступа к объектам базы данных (таблицам, представлениям, процедурам и другим ресурсам).

-Отвечает за авторизацию, то есть проверку наличия соответствующих привилегий на доступ к различным элементам базы данных.

Логин управляет доступом к серверу базы данных, а пользователь управляет доступом к самой базе данных и ее содержимому.

2. Какие системные роли позволяют создавать новые базы данных?

Создание новых баз данных в большинстве популярных СУБД контролируется специальными ролями и привилегиями.

dbcreator (SQL Server) — встроенная роль, позволяющая пользователям создавать, удалять и изменять базы данных на сервере.

CREATDB (PostgreSQL) — привилегия, дающая право пользователю создавать новую базу данных.

SYSDBA (Oracle Database) — администраторская роль, обладающая всеми полномочиями, включая создание баз данных.

SUPERUSER (PostgreSQL) — аналогично SYSDBA в Oracle, супер-пользователь обладает неограниченными возможностями управления базой данных.

3. Как ограничить доступ пользователя к определенным столбцам таблицы?

Доступ пользователя к отдельным столбцам таблицы можно ограничить несколькими способами, зависящими от конкретной системы управления базами данных (СУБД).

1. Использование GRANT и REVOKE (на уровне столбцов)

Многие реляционные базы данных поддерживают управление привилегиями на уровне столбцов. Привилегии можно назначать индивидуально для каждого столбца таблицы.

2. Использование представлений (Views)

Создать представление, которое возвращает только разрешённые столбцы таблицы, и предоставить пользователю доступ к этому представлению, а не к оригинальной таблице.

3. Настройка Row-level Security (RLS) и Column-level Security

Некоторые продвинутые СУБД, такие как SQL Server и PostgreSQL, имеют поддержку row-level security и column-level security, что позволяет ещё точнее контролировать доступ.

4. Автоматизированные механизмы фильтрации (Policy-Based Management)

Некоторые СУБД предлагают встроенные механизмы политики безопасности, позволяющие автоматизировать ограничение доступа к столбцам на основании условий.

Ограничив доступ пользователя к определённым столбцам, вы защищаете конфиденциальные данные и снижаете риски случайного или намеренного повреждения данных пользователями с недостаточными полномочиями. Это важный аспект обеспечения конфиденциальности и целостности данных.

4. Что такое принцип минимальных привилегий и почему он важен?

Принцип минимальных привилегий гласит, что каждому пользователю или процессу должна назначаться минимальная совокупность прав, необходимых исключительно для выполнения возложенных на него обязанностей. То есть каждая учетная запись получает ровно столько полномочий, сколько ей действительно нужно для нормальной работы, но не больше.

Безопасность. Если у пользователя есть избыточные полномочия, злоумышленники смогут воспользоваться этими дополнительными правами в случае компрометации аккаунта. Минимальные привилегии снижают вероятность злоупотреблений и ущерба от взлома.

Простота администрирования. Учетные записи с ограниченными правами требуют меньше внимания администратора и упрощают мониторинг потенциальных угроз.

Соответствие стандартам. Принцип минимальных привилегий является частью многих отраслевых стандартов и лучших практик кибербезопасности