

# Лекция №3

каф. КИБЭВС  
И.В. Горбунов

# Управление доступом

**Доступ к информации** – ознакомление с информацией и ее обработка.

**Субъект доступа** – лицо или процесс, действия которого регламентируются правилами разграничения доступа.

**Объект доступа** – информационная единица АС, доступ к которой регламентируется правилами разграничения доступа.

(объектом может быть все что угодно, содержащее конечную информацию: база данных, таблица, строка, столбец и т.д.)

## **Правила разграничения доступа -**

совокупность правил, регламентирующих права субъектов доступа к объектам доступа.

**Санкционированный доступ** - доступ к информации в соответствии с правилами разграничения доступа.

**Несанкционированный доступ** - доступ к информации, нарушающий правила разграничения доступа в любом проявлении или реализации.

**Идентификатор доступа** - уникальный признак объекта или субъекта доступа.

**Пароль** - идентификатор субъекта, который является его секретом.

# Процесс получения доступа пользователя к БД в СУБД



В случае разрыва соединения:  
транзакция откатывается,  
пользователь переподключается.

# Модели разграничения доступа



# Дискреционное управление доступам — разграничение доступа между поименованными субъектами и поименованными объектами

Объект Субъект	O <sub>1</sub>	O <sub>2</sub>	O <sub>3</sub>	O <sub>4</sub> (Printer)
S <sub>1</sub>	read			
S <sub>2</sub>				Print
S <sub>3</sub>		Read	Execute	
S <sub>4</sub>	read write		read write	



## Ролевая управление доступом

Ролевое разграничение доступа представляет собой развитие политики дискреционного разграничения доступа, при этом права доступа субъектов системы на объекты группируются с учетом специфики их



## **Мандатное управление доступом**

предполагает назначение объекту метки секретности, а субъекту – уровня допуска.

Доступ субъектов к объектам в этой модели снижает производительность компьютерной системы, т.к. проверка прав доступа должна производиться при любой операции с объектом, а не только при его открытии, как в дискреционной модели.

## Управление пользователями

После проектирования логической структуры базы данных, связей между таблицами, ограничений целостности и других структур необходимо определить круг **пользователей**, которые будут иметь *доступ* к базе данных.

В большинстве СУБД двухуровневая схема ограничения доступа к данным:

1. создание учетной записи пользователя, для подключения к серверу (но не получения прав)
2. определение полномочий (уровней доступа) относительно каждой БД в СУБД.

Роль – «объект» СУБД, определяющий уровень доступа субъектов к объектам СУБД.

Роль делиться на 2 группы:

1. на уровне сервера;
2. на уровне БД.

Роль – «объект» СУБД, определяющий уровень доступа субъектов к объектам СУБД.

Роль делиться на 2 группы:

1. на уровне сервера:

- аутентификация;
- учетная запись;
- встроенная роль сервера.

2. на уровне БД:

- пользователь БД;
- фиксированная роль БД;
- пользовательская роль БД.



Для **создания пользователя** следует предпринять следующие шаги:

1. Создать в базе данных *учетную запись пользователя*, указав для него *пароль* и принятое по умолчанию имя базы данных
2. Добавить этого *пользователя* во все необходимые базы данных.
3. Предоставить ему в каждой базе данных соответствующие *привилегии* .



## Стандартные процедуры

*Создание новой учетной записи* может быть произведено с помощью системной хранимой процедуры:

```
sp_addlogin
```

```
[@login=] 'учетная_запись'
```

```
[, [@password=] 'пароль']
```

```
[, [@defdb=] 'база_данных_по_умолчанию']
```

*Пользователь, который создает объект в базе данных (таблицу, хранимую процедуру, просмотр), становится его **владельцем**.*

*Владелец объекта (database object owner dbo) имеет все **права доступа** к созданному им объекту.*

*Чтобы **пользователь** мог создать объект, **владелец** базы данных (dbo) должен предоставить ему соответствующие **права**.*

*Полное имя создаваемого объекта включает в себя имя создавшего его **пользователя** .*

Передача *прав* владения от  
одного *пользователя* другому с помощью  
процедуры:

sp\_changeobjectowner

[@objname=] 'имя\_объекта'

[@newowner=] 'имя\_владельца'

*Роль* позволяет объединить в одну группу *пользователей*, выполняющих одинаковые функции.

В SQL Server реализовано два вида стандартных *ролей*: на уровне сервера и на уровне баз данных.

Фиксированные роли сервера:

- `sysadmin` с *правом* выполнения любых функций SQL-сервера.

Фиксированные *роли* базы данных:

- `db_owner` с *правом* полного доступа к базе данных;
- `db_accessadmin` с *правом* добавления и удаления *пользователей*.

**Роли** базы данных позволяют объединять *пользователей* в **одну административную единицу** и работать с ней как с обычным *пользователем*.

Можно назначить *права доступа* к объектам базы данных для конкретной *роли*, при этом автоматически все члены этой *роли* наделяются *одинаковыми правами*.



создание новой *роли*:

sp\_addrole

[@rolename=] 'имя\_роли'

[, [@ownername=] 'имя\_владельца']

добавление *пользователя* к *роли*:

sp\_addrolemember

[@rolename=] 'имя\_роли',

[@membername=] 'имя\_пользователя'



удаление *пользователя* из роли:

```
sp_droprolemember
```

```
[@rolename=] 'имя_роли',
```

```
[@membername=] 'имя_пользователя'
```

удаление *роли*:

```
sp_droprole
```

```
[@rolename=] 'имя_роли'
```

**Спасибо за внимание!!!**