* + - 1. **Показать и объяснить, какой режим аутентификации используется для экземпляра SQL Server.**

Режим аутентификации Windows (Windows Authentication): Этот режим использует учетные записи Windows для аутентификации пользователей.

Режим аутентификации SQL Server (SQL Server Authentication): В этом режиме пользователи аутентифицируются с использованием учетных записей SQL Server. При этом требуется указание имени пользователя и пароля для подключения к серверу. Учетные записи хранятся в базе данных SQL Server и могут использоваться для ограничения доступа к данным.

***Схема безопасности в SQL Server*** — это организация и управление правами доступа к базе данных и ее объектам. Включает в себя следующие компоненты:

Принципалы (Principals): Принципалы в SQL Server представляют субъектов, которые могут быть аутентифицированы и авторизованы для доступа к базе данных.

Роли (Roles): Роли представляют группы принципалов, которым можно назначать права доступа и разрешения на уровне базы данных.

Разрешения (Permissions): Разрешения определяют, какие операции могут выполняться на объектах базы данных.

Схемы (Schemas): Схемы представляют собой логические контейнеры для объектов базы данных, таких как таблицы, представления и процедуры.

В контексте безопасности SQL Server:

***Принципалы*** (Principals): Принципалы представляют пользователей, группы или учетные записи, которым разрешен доступ к базе данных. Они могут быть аутентифицированы с помощью учетных данных Windows или SQL Server и могут выполнять операции в базе данных.

***Подзащитные*** (Securables): Подзащитные представляют объекты базы данных, к которым могут быть применены разрешения. Это могут быть таблицы, представления, хранимые процедуры, функции и другие объекты базы данных. Подзащитные определяют, на какие объекты могут быть назначены разрешения и какие операции можно выполнять с этими объектами.

* + - 1. **Создать необходимые учетные записи, роли и пользователей. Объяснить назначение привилегий.**

Есть 2 типа учетных записей логины и пользователи:

* + - * + Логин – представляют собой идентификаторы и учетные данные, используемые для аутентификации пользователей при подключении к серверу. Логины могут быть связаны с учетными записями Windows (Windows logins) или могут быть определены независимо в SQL Server (SQL Server logins).
        + Пользователь – связаны с конкретными базами данных и представляют собой учетные записи, которые имеют доступ и разрешения на выполнение операций внутри базы данных. Пользователи могут быть созданы на основе существующих логинов или могут быть определены независимо

ЛОГИНЫ: -- проверить: Security => Logins => sa

ПОЛЬЗОВАТЕЛИ: -- проверить: Databases => «Имя бд» => Users

***Встроенные роли сервера***: Security => Server Roles

Для просмотра привилегий: EXEC sp\_srvrolepermission 'sysadmin';

***Встроенные роли бд***: Databases => «Имя бд» => Roles => Database Roles

Для просмотра привилегий: SELECT \* FROM sys.database\_permissions WHERE grantee\_principal\_id = DATABASE\_PRINCIPAL\_ID('public');

* + - 1. Продемонстрируйте заимствование прав для любой процедуры в базе данных.

***Имперсонификация*** – разрешение заимствовать права у другого пользователя для выполнения определенного блока кода.

SETUSER остается в силе, пока не будет выполнен другой оператор SETUSER или пока текущая база данных не будет изменена с помощью оператора USE.

***Цепочка владения*** – Ownership Chain

SQL Server пропускает проверку привилегий для улучшения производительности для двух объектов с одинаковым владельцем.

* + - 1. Создать для экземпляра SQL Server объект аудита.

Аудит – отслеживание и протоколирование событий, происходящих на сервере.

Аудит SQL Server:

\*аудит сервера для событий на уровне сервера

\*аудит бд для событий на уровне бд

Журнал аудита:

\*файл файловой системы

\*журнал windows application log

\*windows security log

**Схема:**

**Серверный аудит -> спецификация аудита (какие события и действия записываются) -> аудиторский журнал (место хранения) -> запуск аудита**

**Куда записывается?**

**Аудиторский журнал базы данных (таблица в бд), аудиторский журнал сервера (файл).**

--проверить: безопасность => аудит (свойства)

* + - 1. Задать для серверного аудита необходимые спецификации.

Типы действия аудита:

|  |  |
| --- | --- |
| database\_change\_group | Изменение баз данных |
| database\_object\_change\_group | Изменения над объектами базы данных |
| login\_change\_password\_group | Изменения паролей |
| server\_state\_change\_group | Модификация служб |
| successful\_login\_group | Успешный вход |
| failed\_login\_group | Неудачная попытка входа |
| logout\_group | Выход пользователя из SQL Server |

* + - 1. Запустить серверный аудит, продемонстрировать журнал аудита.

--запуск: безопасность => аудит => правой кнопкой мыши (включить аудит)

--журнал аудита: безопасность => аудит => правой кнопкой мыши (посмотреть журналы)

* + - 1. Создать необходимые объекты аудита БД + создать спецификацию + запустить аудит + журнал

--сделать: БД lab10 => безопасность => спецификации аудита бд => создать

(insert, object, выбираем имя объекта (таблица), выбираем имя участника (dbo))

10. Остановить аудит БД и сервера.

Все отключить

11. Создать для экземпляра SQL Server ассиметричный ключ шифрования. Зашифровать и расшифровать данные при помощи этого ключа.

**Шифрование:**

**\* Service Master Key – шифрует мастер-ключи БД**

**\* Database Master Key – шифрует симметричные ключи**

**\* Симметричные ключи – шифруют данные**

**Схема шифрования:**

**Мастер-ключ (Master Key): используется для защиты других ключей и секретных данных, связанных с шифрованием. Он обычно создается на основе пароля или сертификата.**

**Сертификаты (Certificates): Сертификаты используются для защиты мастер-ключа и других ключей шифрования. Сертификаты могут быть созданы в базе данных или импортированы из внешних источников. Они обеспечивают цифровую подпись и проверку подлинности данных и ключей.**

**Асимметричные ключи (Asymmetric Keys): Асимметричные ключи используются для шифрования данных и обмена ключами с другими участниками. Они основаны на асимметричной криптографии и состоят из открытого и закрытого ключей.**

**Симметричные ключи (Symmetric Keys): Симметричные ключи используются для фактического шифрования и расшифрования данных. Они используют один и тот же ключ для шифрования и расшифрования и могут быть созданы с использованием паролей, сертификатов или асимметричных ключей.**

**Ключи шифрования данных (Data Encryption Keys): Ключи шифрования данных используются для защиты конкретных данных в базе данных. Они шифруются и расшифровываются с помощью симметричных ключей.**

**Представления, связанные с шифрованием:**

**sys.certificates, sys.symmetric\_keys**

--проверить что ключ создан: БД => безопасность => ассиметричные ключи

13. Создать для экземпляра SQL Server сертификат.

Сертификат – подписанная цифр. подписью инструкция, кот. связывает значение откр. ключа с id юзера, у-ва или службы, кот. имеет соотв. закрытый ключ.

Содержит:

* Откр. ключ субъекта
* Идентификационные данные субъекта + поставщика (имя и адрес эл.почты)
* Срок действия (valid from / valid to)
* Цифр. подпись поставщика

--проверить: БД => безопасность => сертификаты

15. Создать для экземпляра SQL Server симметричный ключ шифрования данных.

1. Создаем симм. ключ SKey для шифрования симм. ключа  
2. Открыть его

3. Создаем симм. ключ SData для шифрования данных

4. Открыть его

Create, alter, drop, open, close, close all symmetric keys;

17. Продемонстрировать прозрачное шифрование базы данных.

Tde – transparent data encryption – прозрачное шифрование д-х

исп. для шифрования данных и файлов журналов перед их записью на диск и расшифр. перед их возвратом в приложение.

Это вып. на уровне [SQL](https://ru.wikipedia.org/wiki/SQL). В последующих резервных копиях файлов БД данные будут зашифрованы.

Чтобы выполнить прозрачное шифрование необходимо:

* + - * 1. Создать главный ключ
        2. Создать или получить сертификат, защищенный главным ключом
        3. Создать ключ шифрования бд и защитить его с помощью сертификата
        4. Задайте ведение шифрования бд

=> обнуляется оставшаяся часть вирт. журнала транзакций и принудительно начинается новый

Это гарантирует, что после включения шифрования бд в журналах транзакций не останется простого текста.

19. Продемонстрировать применение ЭЦП при помощи сертификата.

SignByCert (id сертификата, текст, пароль)

VerifySignedByCert (certificate\_ID, plaintext, signature)

SignByAsymKey (asym\_key\_ID, plaintext, password)

VerifySignedByAsymKey (asym\_key\_ID, plaintext, signature)