**НЕГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ ЧАСТНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«МОСКОВСКИЙ ФИНАНСОВО-ПРОМЫШЛЕННЫЙ УНИВЕРСИТЕТ «СИНЕРГИЯ»**

|  |  |  |
| --- | --- | --- |
| **Факультет/Институт** |  | Информационных технологий |
|  |  | (наименование факультета/ Института) |
| **Направление/специальность** |  | Разработка, сопровождение и обеспечение безопасности информационных систем |
| **подготовки:** |  | (код и наименование направления /специальности подготовки) |
| **Форма обучения:** |  | очная |
|  |  | (очная, очно-заочная, заочная) |
|  |  |  |

**Лабораторный практикум № 9**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **по дисциплине** |  | Администрирование информационных систем | | |
|  | | |  | (наименование дисциплины) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Обучающийся** |  | **Низомов Рустам С** |  |  |
|  |  | (ФИО) |  | (подпись) |
| **Группа** |  | **Вбио-305-рсоб** |  |
|  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Преподаватель** |  | Сибирев Иван Валерьевич |  |  |
|  |  | (ФИО) |  | (подпись) |

**Москва 2024 г**

Проектирование защиты баз данных SQL Server: Роли, разрешения и шифрование

1. Создание пользовательских ролей и предоставление доступа к объектам

\* Создание роли:

CREATE ROLE [Имя роли]

\* Пример: CREATE ROLE DataReaders

\* Предоставление разрешений:

GRANT SELECT ON [Имя таблицы] TO [Имя роли]

\* Пример: GRANT SELECT ON dbo.Customers TO DataReaders

\* Другие разрешения: INSERT, UPDATE, DELETE, EXECUTE и т.д.

2. Определение объектов базы данных только для чтения

\* Создание представления:

CREATE VIEW [Имя представления] AS

SELECT \* FROM [Имя таблицы]

\* Предоставление доступа к представлению:

GRANT SELECT ON [Имя представления] TO [Имя роли]

\* Ограничение модификации базовой таблицы:

DENY UPDATE, DELETE ON [Имя таблицы] TO [Имя роли]

3. Определение изменяемых объектов и частоты корректировки

\* Анализ данных: Определить, какие данные часто изменяются, какие редко.

\* Разделение данных: Разделить данные на группы по частоте изменения.

\* Создание отдельных таблиц или схем: Для каждой группы создать отдельную таблицу или схему.

\* Предоставление дифференцированных разрешений: Предоставить разные уровни доступа для разных групп данных.

4. Создание пользователей и добавление в роли

\* Создание пользователя:

CREATE USER [Имя пользователя] FOR LOGIN [Имя входа Windows]

\* Добавление пользователя в роль:

EXEC sp\_addrolemember 'Имя роли', 'Имя пользователя'

Пример схемы безопасности

-- Создание ролей

CREATE ROLE DataReaders;

CREATE ROLE DataWriters;

CREATE ROLE DataOwners;

-- Создание таблицы

CREATE TABLE Customers (

CustomerID int PRIMARY KEY,

CompanyName nvarchar(40),

ContactName nvarchar(30),

ContactTitle nvarchar(30),

Address nvarchar(60),

City nvarchar(15),

Region nvarchar(15),

PostalCode nvarchar(10),

Country nvarchar(15)

);

-- Предоставление разрешений

GRANT SELECT ON Customers TO DataReaders;

GRANT INSERT, UPDATE, DELETE ON Customers TO DataWriters;

GRANT ALL ON Customers TO DataOwners;

-- Создание пользователей

CREATE USER User1 FOR LOGIN 'Domain\User1';

CREATE USER User2 FOR LOGIN 'Domain\User2';

-- Добавление пользователей в роли

EXEC sp\_addrolemember 'DataReaders', 'User1';

EXEC sp\_addrolemember 'DataWriters', 'User2';

Шифрование данных

\* Шифрование на уровне столбца: Зашифровать отдельные столбцы, содержащие конфиденциальные данные.

\* Шифрование на уровне базы данных: Зашифровать всю базу данных или отдельные файлы данных.

\* Always Encrypted: Технология для шифрования данных "в покое" и "в движении".

Дополнительные меры безопасности

\* Аудит: Отслеживать действия пользователей и выявлять подозрительную активность.

\* Двухфакторная аутентификация: Повысить уровень безопасности учетных записей.

\* Сегментация сети: Разделить сеть на зоны для ограничения доступа.

\* Регулярное обновление: Обновлять SQL Server и операционную систему.

Проектирование схемы

\* Нормализация данных: Уменьшить избыточность и зависимости между данными.

\* Индексация: Создать индексы для ускорения поиска данных.

\* Партиционирование: Разделить большие таблицы на более мелкие для повышения производительности.

Важно:

\* Принцип наименьших привилегий: Предоставлять пользователям только те разрешения, которые необходимы для выполнения их задач.

\* Регулярный обзор: Регулярно пересматривать и обновлять схему безопасности.

\* Документация: Создавать подробную документацию по схеме безопасности.

Дополнительные соображения:

\* Безопасность на уровне сети: Использовать брандмауэры и VPN для защиты базы данных.

\* Безопасность приложений: Уделять внимание безопасности приложений, взаимодействующих с базой данных.

\* Безопасность резервных копий: Защищать резервные копии от несанкционированного доступа