Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и кибербезопасности

Высшая школа программной инженерии

Отчет по лабораторной работе

2.1 «Управление доступом»

по дисциплине

«Системы управления базами данных»

Выполнила

Студентка гр. 5130904/10101 Никифорова Е. А.

Руководитель Прокофьев О. В.

Санкт-Петербург

2024

Постановка задачи

Целью практического задания является освоение работы с представлениями и другими способами управления доступом. При выполнении задания необходимо:

* Создать пользователя test и выдать ему доступ к базе данных.
* Составить и выполнить скрипты присвоения новому пользователю прав доступа к таблице, созданной в практическом задании №1.4. При этом права доступа к различным таблицам должны быть различными, а именно:
  + По крайней мере, для одной таблицы новому пользователю присваиваются права SELECT, INSERT, UPDATE в полном объеме.
  + По крайней мере, для одной таблицы новому пользователю присваиваются права SELECT и UPDATE только избранных столбцов.
  + По крайней мере, для одной таблицы новому пользователю присваивается только право SELECT.
* Составить SQL-скрипты для создания нескольких представлений, которые позволяли бы упростить манипуляции с данными или позволяли бы ограничить доступ к данным, предоставляя только необходимую информацию.
* Присвоить новому пользователю право доступа (SELECT) к одному из представлений
* Создать стандартную роль уровня базы данных, присвоить ей право доступа (UPDATE на некоторые столбцы) к одному из представлений, назначить новому пользователю созданную роль.
* Выполнить от имени нового пользователя некоторые выборки из таблиц и представлений. Убедиться в правильности контроля прав доступа.
* Выполнить от имени нового пользователя операторы изменения таблиц с ограниченными правами доступа. Убедиться в правильности контроля прав доступа.
* Используя psql выполнить макрокоманду /dp и описать флаги доступа

Решение:

psql -U postgres -d lab2

CREATE USER test PASSWORD 'pass';

GRANT ALL ON DATABASE lab2 TO test;

|  |  |  |
| --- | --- | --- |
| Ввод –U postgres | Ввод –U test | Вывод |
| GRANT SELECT, INSERT, UPDATE ON table1 to test; |  | GRANT |
|  | psql -U test -d lab2 | Пароль пользователя test:  psql (16.2)  Введите "help", чтобы получить справку. |
|  | select \* from table1; | id | name | age  ----+------+-----  (0 строк) |
|  | INSERT INTO table1 ( name, age) VALUES ('John', 25); | INSERT 0 1 |
|  | select \* from table1; | id | name | age  ----+------+-----  1 | John | 25  (1 строка) |
|  | UPDATE table1  SET name = 'Alice', age = 30 WHERE id = 1; | UPDATE 1 |
|  | select \* from table1; | id | name | age  ----+-------+-----  1 | Alice | 30  (1 строка) |
|  |  |  |
| GRANT SELECT(age), UPDATE(age) ON table2 to test; |  | GRANT |
|  |  |  |
|  |  |  |
|  | INSERT INTO table2 ( name, age) VALUES ('John', 25); | INSERT 0 1 |
| select \* from table2; |  | id | name | age  ----+------+-----  1 | John | 25  (1 строка) |
|  | select \* from table2; | ОШИБКА: нет доступа к таблице table2 |
|  | SELECT age FROM table2; | age  -----  5  (1 строка) |
|  | update table2 set age=35; | UPDATE 1 |
| select \* from table2; |  | id | name | age  ----+------+-----  1 | John | 35  (1 строка) |
|  |  |  |
| GRANT SELECT ON table3 to test; |  | GRANT |
|  | select \* from table3; | id | name | age  ----+------+-----  (0 строк) |
|  | INSERT INTO table3 ( name, age) VALUES ('John', 25); | ОШИБКА: нет доступа к таблице table3 |
| INSERT INTO table3 ( name, age) VALUES ('John', 25);  select \* from table3; |  | id | name | age  ----+------+-----  1 | John | 25  (1 строка) |
|  | update table3 set age=35 where name = 'John'; | ОШИБКА: нет доступа к таблице table3 |
|  |  |  |
| CREATE OR REPLACE VIEW view\_ages AS  SELECT id, age FROM table1; |  | CREATE VIEW |
| GRANT SELECT ON view\_ages to test; |  | GRANT |
|  |  |  |
| CREATE ROLE age\_updater LOGIN; |  | CREATE ROLE |
| GRANT UPDATE ON view\_ages TO age\_updater; |  | GRANT |
| GRANT age\_updater to test; |  | GRANT ROLE |
|  | select \* from view\_ages; | id | age  ----+-----  1 | 30  (1 строка) |
|  | update view\_ages set age = age + 1; | UPDATE 1 |
|  | select \* from view\_ages; | id | age  ----+-----  1 | 31  (1 строка) |
|  | INSERT INTO view\_ages (age) VALUES (100); | ОШИБКА: нет доступа к представлению view\_ages |
|  |  |  |

Вывод:

В ходе лабораторной работы были изучены и протестированы способы управления доступом, такие как работа с представлениями, созданием ролей, предоставление различных привилегий.