

Database-System Experiment-8

21307289 刘森元

1. 实验目的

熟悉数据库用户管理和权限管理，能够使用SQL语句来向用户授予和收回权限。

2. 实验环境

Macbook Pro 2021 (Apple M1 Pro)

macOS Ventura 13.5.2

PostgreSQL 15.4 (Homebrew)

zsh 5.9

3. 实验内容

1. 使用GRANT语句对用户授权，对单个用户和多个用户授权，或使用保留字PUBLIC对所有用户授权。对不同的操作对象包括数据库、视图、基本表等进行不同权限的授权。
2. 使用WITH GRANT OPTION子句授予用户传播该权限的权利。
3. 在授权时发生循环授权，考察DBS能否发现这个错误。如果不能，结合取消权限操作，查看DBS对循环授权的控制。
4/ 使用REVOKE子句收回授权，取消授权的级联反应。

4. 实验步骤

完整代码见附件 [Experiment-8.sql](#)

使用命令 `psql --dbname=postgres --file=Experiment-8.sql --echo-errors --quiet` 以运行脚本。

重建数据库

由于本次试验涉及到增删，若重复操作初始数据库内容会不一致，所以该项操作是必要的。

```
-- 重建数据库
DROP DATABASE "school";
DROP USER USER1;
DROP USER USER2;
DROP USER USER3;
CREATE DATABASE "school";
\c school
\cd '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-8'
\i STUDENTS.sql
```

```
\i TEACHERS.sql
\i COURSES.sql
\i CHOICES.sql

-- 创建用户
CREATE USER USER1 PASSWORD '111';
CREATE USER USER2 PASSWORD '222';
CREATE USER USER3 PASSWORD '333';
```

课内实验

1. 授予所有用户对表COURSES的查询权限

```
GRANT SELECT ON COURSES TO USER1, USER2, USER3;
```

2. 授予USER1对表STUDENTS插入和更新的权限，但不授予删除权限，并且授予USER1传播这两个权限的权利

```
GRANT INSERT, UPDATE ON STUDENTS TO USER1 WITH GRANT OPTION;
```

3. 允许USER2在表CHOICE中插入元组，更新的SCORE列，可以选取除了SID以外的所有列

```
GRANT INSERT, UPDATE (score) ON CHOICES TO USER2;
GRANT SELECT (no, tid, cid, score) ON CHOICES TO USER2;
```

4. USER1授予USER2对表STUDENTS插入和更新的权限，并且授予USER2传播插入操作的权利

```
\c - user1
GRANT INSERT, UPDATE ON STUDENTS TO USER2;
GRANT INSERT ON STUDENTS TO USER2 WITH GRANT OPTION;
```

5. 收回对USER1对表COURSES查询权限的授权

```
\c - qiu_nangong
REVOKE SELECT ON COURSES FROM USER1;
```

6. 由上面2. 和4. 的授权, 再由USER2对USER3授予表STUDENTS插入和更新的权限, 并且授予USER3传播插入操作的权利。这时候, 如果由USER3对USER1授予表STUDENTS的插入和更新权限是否能得到成功?如果能够成功, 那么如果由USER2取消USER3的权限, 对USER1会有什么影响?如果再由DBA取消USER1的权限, 对USER2有什么影响?

```
\c - user2
GRANT INSERT, UPDATE ON STUDENTS TO USER3;
GRANT INSERT ON STUDENTS TO USER3 WITH GRANT OPTION;
```

根据上述授权, USER2授予USER3对STUDENTS表的插入和更新权限, 并且授予USER3传播插入操作的权利。在这种情况下, 如果USER3尝试向USER1授予对STUDENTS表的插入和更新权限, 这将不会成功。只有表的所有者或具有适当管理权限的用户才能授予权限。

如果USER2取消了USER3在STUDENTS表上的权限, 这不会对USER1的权限产生任何影响。USER1仍然保留其与STUDENTS表相关的原始权限。

如果DBA取消了USER1的权限, 这将影响USER2。由于USER2依赖于USER1传播的权限, 一旦USER1的权限被取消, USER2将无法再传播这些权限给其他用户。USER2将失去对STUDENTS表的插入和更新权限的传播能力。

自我实践

1. 授予所有用户对表STUDENTS的查询权限

```
GRANT SELECT ON STUDENTS TO USER1, USER2, USER3;
```

2. 授予所有用户对表COURSES的查询和更新权限

```
GRANT SELECT, UPDATE ON COURSES TO USER1, USER2, USER3;
```

3. 授予USER1对表TEACHERS的查询, 更新工资的权限, 且允许USER1可以传播这些权限

```
GRANT SELECT, UPDATE (salary) ON TEACHERS TO USER1 WITH GRANT OPTION;
```

4. 授予USER2对表CHOICES的查询, 更新成绩的权限

```
GRANT SELECT, UPDATE (score) ON CHOICES TO USER2;
```

5. 授予USER2对表TEACHERS的除了工资之外的所有信息的查询

```
GRANT SELECT (tid, tname, email) ON TEACHERS TO USER2;
```

6. 由USER1授予USER2对表TEACHERS的查询权限和传播的此项权限的权利

```
\c - user1
GRANT SELECT ON TEACHERS TO USER2 WITH GRANT OPTION;
```

7. 由USER2授予USER3对表TEACHERS的查询权限，和传播的此项权限的权利。再由USER3授予USER2上述权限，这样的SQL语句能否成功得到执行？

```
\c - user2
GRANT SELECT ON TEACHERS TO USER3 WITH GRANT OPTION;
\c - user3
GRANT SELECT ON TEACHERS TO USER2 WITH GRANT OPTION;
```

该语句不能成功执行，提示错误

```
psql:Experiment-8.sql:73: ERROR:  grant options cannot be granted back to your own
grantor
psql:Experiment-8.sql:73: 语句:  GRANT SELECT ON TEACHERS TO USER2 WITH GRANT
OPTION;
```

这是因为不能循环授权

8. 取消USER1对表STUDENTS的查询权限，考虑由USER2的身份对表STUDENTS进行查询，操作能否成功?为什么？

```
\c - qiu_nangong
REVOKE SELECT ON STUDENTS FROM USER1;
\c - user2
SELECT COUNT(*) FROM STUDENTS;
```

该操作不能成功执行，这是因为USER2的查询权限来源于USER1，若USER1被取消权限则其授予权限的用户都将被级联取消。

9. 取消USER1和USER2的关于表COURSES的权限

```
\c - qiu_nangong
REVOKE SELECT, UPDATE ON COURSES FROM USER1, USER2;
```

5. 实验心得

在这个实验中，我学到了如何在数据库中创建用户、授予和撤销权限，以及管理用户之间的权限传播。通过使用GRANT和REVOKE语句，我能够授予和撤销用户对特定表的查询、插入、更新等权限。我还学会了如何使用WITH GRANT OPTION来允许用户传播他们拥有的权限。这对于管理和授权用户非常有帮助。