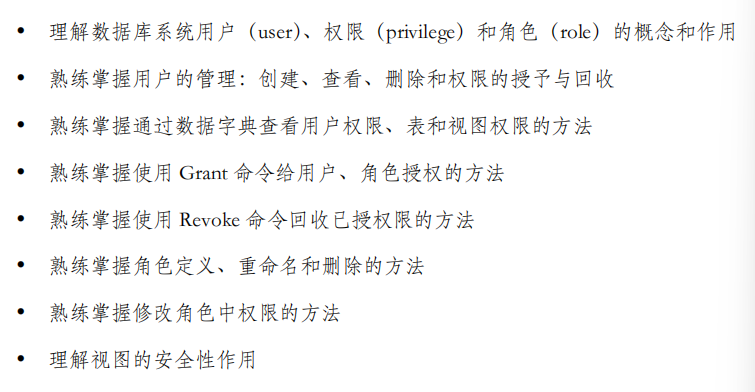


数据库系统课程实验报告

|  |  |
| --- | --- |
| 实验名称： | 数据库的安全性 |
| 实验日期： | 2022/4/21 |
| 实验地点： | 四号楼 |
| 提交日期： | 2022/4/24 |
|  | |
| 学号： | 22920202202879 |
| 姓名： | 陈奕培 |
| 专业年级： | 软工2020级 |
| 学年学期： | 2021-2022学年第二学期 |

1. 实验目的



1. 实验内容和步骤

（1）完成网页内容

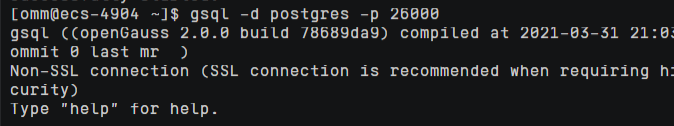
1.切换到omm



2.启动服务

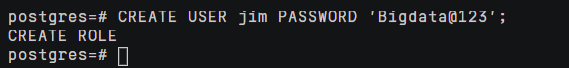


3.连接到数据库

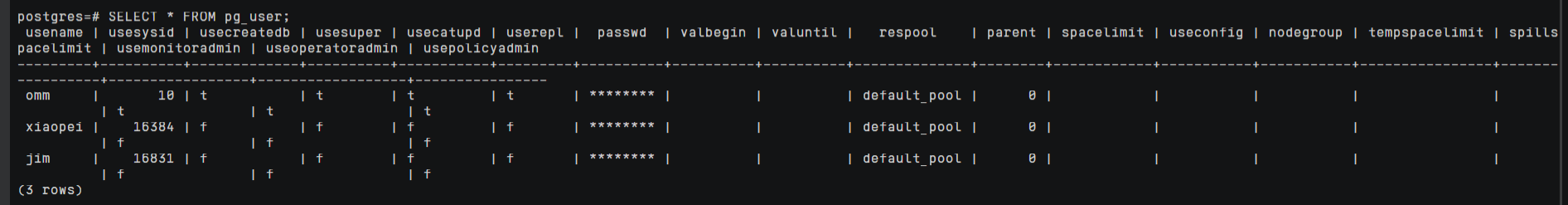


4.用户

4.1新建用户



4.2查看用户列表



4.3为jim追加创建角色权限



4.4删除用户

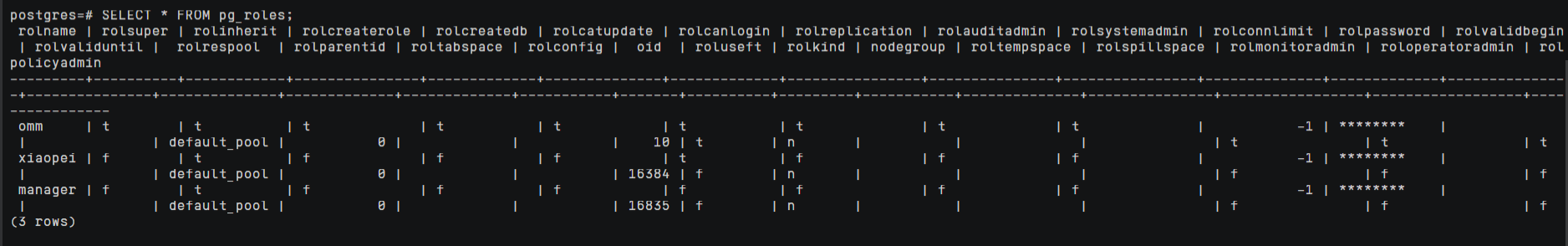


5.角色

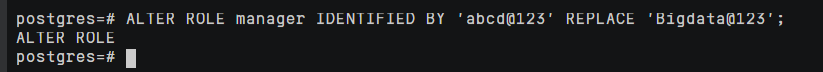
5.1创建角色



5.2查看角色



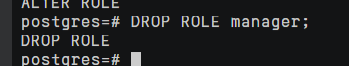
5.3修改角色密码



5.4修改角色manager为系统管理员



5.5删除角色



6.权限设置

6.1创建用户joe



6.2将sysadmin权限授权给joe

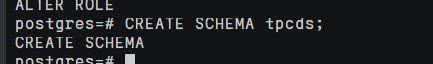


6.3撤销joe的sysadmin权限

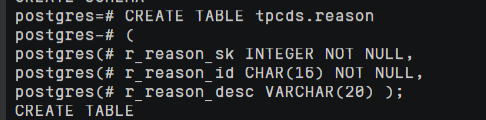


7.将数据库对象授权给角色或用户

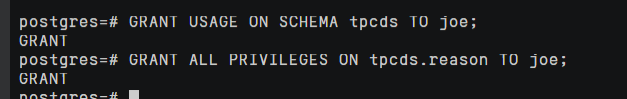
7.1创建tpcds模式



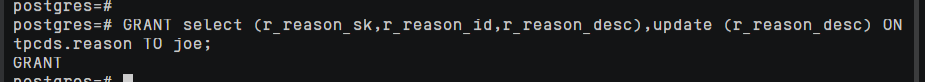
7.2创建表



7.3将模式的使用权限和表的所有权限授予joe



7.4将tpcds.reason表中r\_reason\_sk、r\_reason\_id、r\_reason\_desc列的查询权限，r\_reason\_desc的更新权限授权给joe



7.5将数据库postgres的连接权限授权给用户joe，并给予其在postgres中创建schema的权限，而且允许joe将此权限授权给其他用户



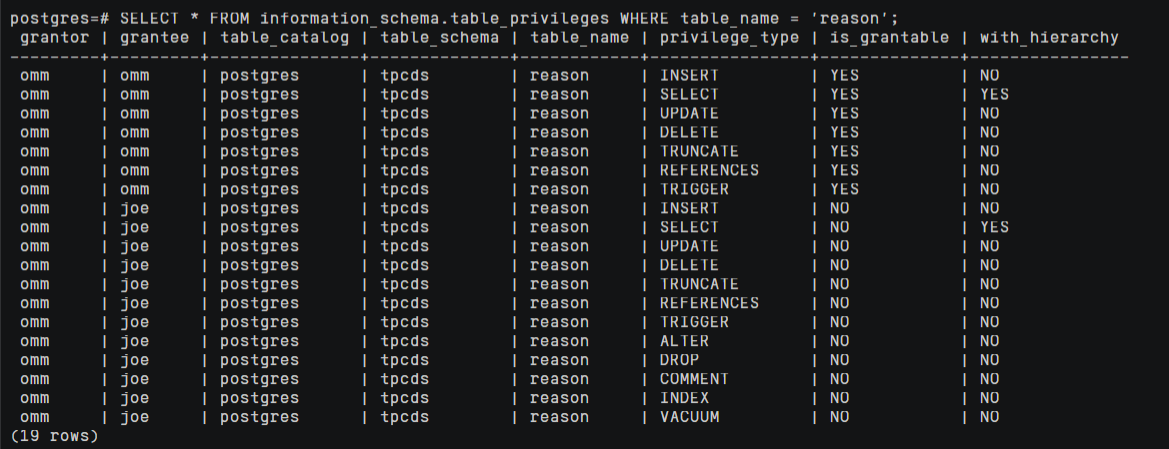
7.6创建角色tpcds\_manager



7.7将模式tpcds的访问权限授权给角色tpcds\_manager，并授予该角色在tpcds下创建对象的权限，不允许该角色中的用户将权限授权给其人

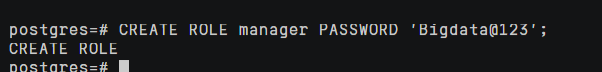


7.8查看表reason的权限



8.将用户或者角色的权限授权给其他用户或者角色

8.1创建角色manager



8.2将joe的权限授权给manager，并允许该角色将权限授权给其他人



8.3创建用户senior\_manager

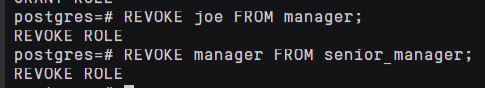


8.4将用户manager的权限授权给该用户



9.权限回收

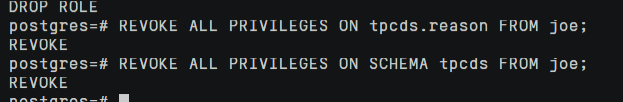
9.1逐步回收manager的权限



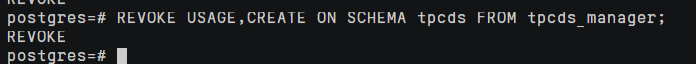
9.2删除manager用户



9.3逐步回收joe权限



9.4逐步回收tpcds\_manager权限



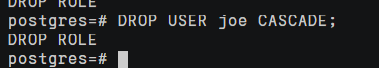
9.5删除tpcds\_manager用户



9.6删除senior\_manager用户

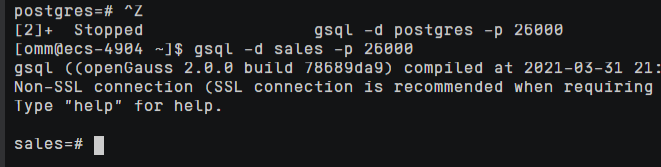


9.7删除joe用户

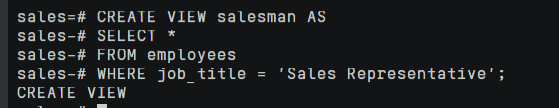


（2）创建视图salesman，该视图只保存employees表中所有job\_title为'Sales Representative'的雇员。

1.切换到sales数据库



2.创建视图

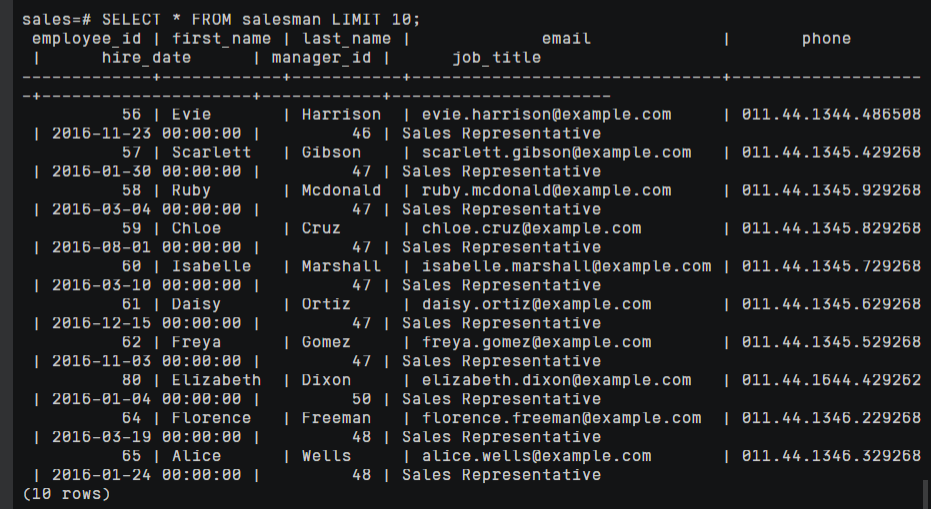


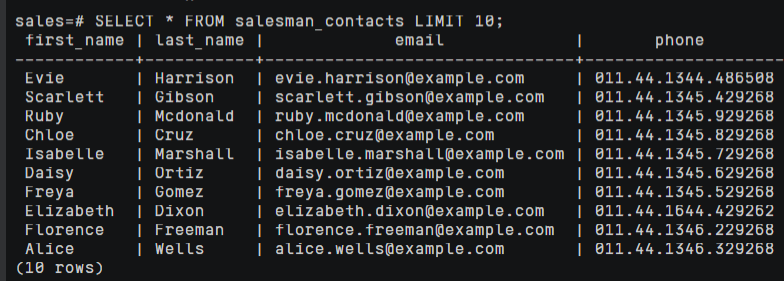
（3）创建基于 salesman 的视图 salesman\_contacts(first\_name,last\_name,email,phone)，该视图存储的

salesman 的联系方式。

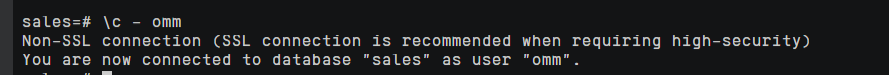


（4）查询视图salesman和salesman\_contacts

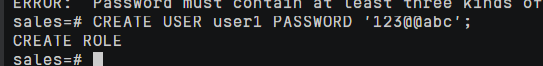




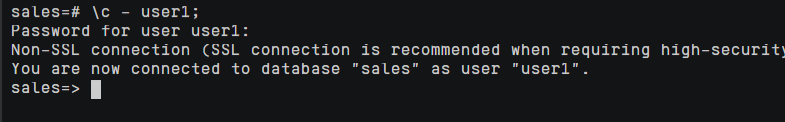
（5）在当前窗口输入命令：\c - omm 切换到 omm 用户



（6）创建新用户 user1



（7）在当前窗口输入命令：\c – user1 切换到 user1 用户

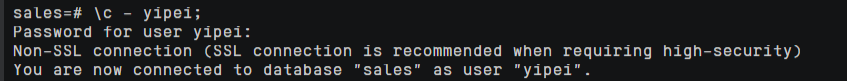


（8）发布查询命令：select \* from salesman\_contacts;观察结果



显示权限不够

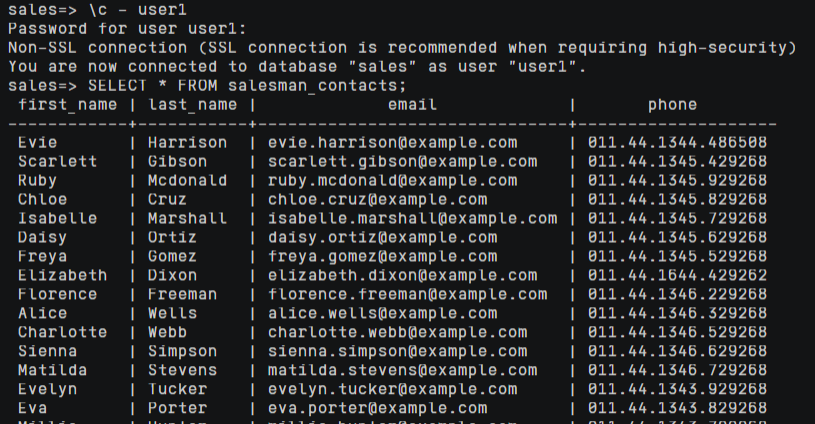
（9）切换到yipei



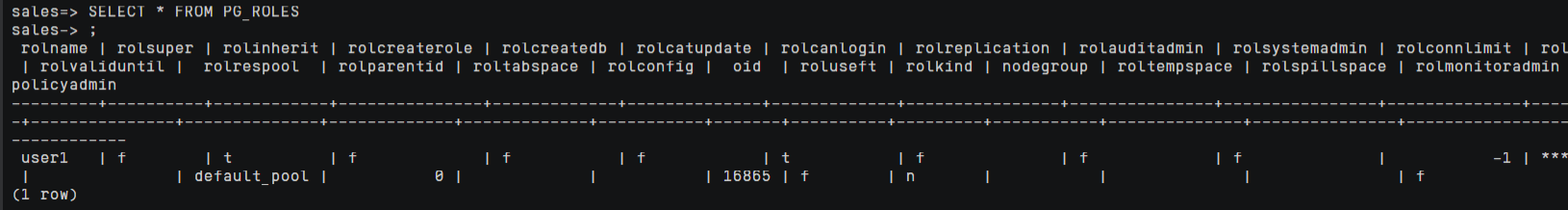
（10）授权

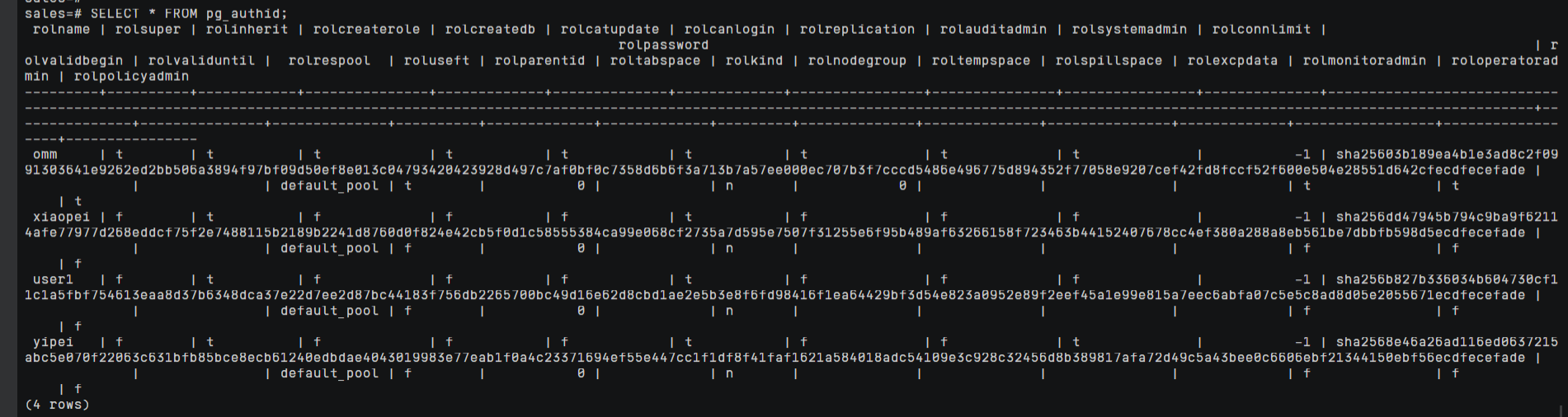


（11）重复7，8



（12）查看与角色、权限相关的系统表和系统视图：pg\_roles，pg\_authid





（13）在完成（1）的基础上，重做教材中的[例 4.1-例 4.13]，因为 openGauss 的语法与教材上的不完全一致，可以通过以上实操加深对 openGauss 安全性控制机制的理解

1.将student表的查询权限授给u1



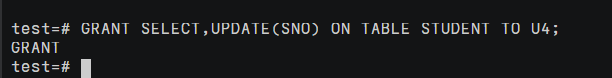
2.对student和course的所有权限授予u2和u3



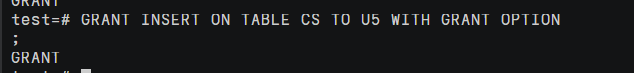
3.对sc表的查询权限授予所有用户



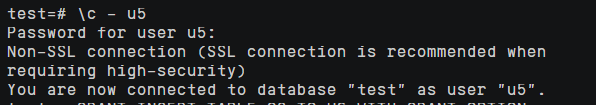
4.将查询student和修改学号的权限授予u4

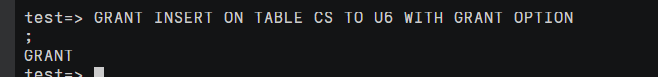


5.把对cs表插入的权限授予u5，并且可以再次授权

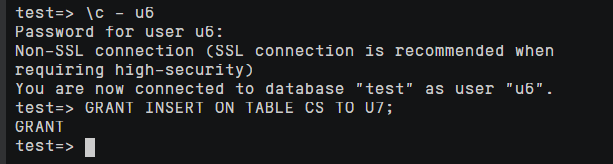


6.切换到u5，授权给u6

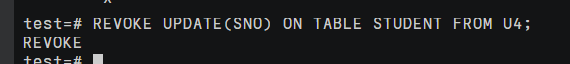




7.切换到u6，授权给u7



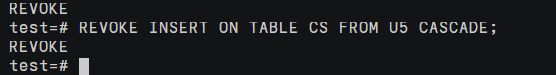
8.把用户u4的权限收回



9.收回对cs的查询权限

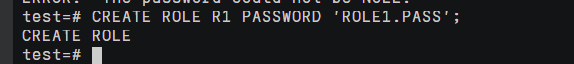


10.把u5的权限收回

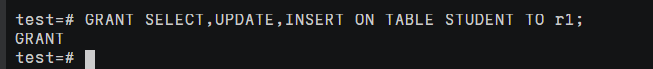


11.创建角色来授权用户

11.1新建角色



11.2授权角色



11.3授权



11.4回收



12.修改权限



13.修改权限



实验思考

* 只有拥有DBA权限的用户才能创建新用户
* 角色是拥有数据库对象和权限的实体，在不同的环境中可以认为是一个或一组用户
* Grant和revoke可以实现角色权限的修改

1. 实验总结

3.1 完成的工作

创建视图

创建用户，角色，对他们赋予不同的权限，收回他们的权限

3.2 对实验的认识

掌握了视图的创建；

新建用户、新建角色；

对用户和角色赋予不同的权限；

把用户的权限赋予角色；

角色的权限赋予用户；

理解视图的安全性作用；

3.3 遇到的困难及解决方法

无