

数据库系统实验报告（四）

课程名称： 数据库系统原理 实验项目名称： SQL 安全性

学生姓名： 刘轩铭 专业： 软件工程 学号： 3180106071

指导老师： 周波 实验日期： 2020 年 4 月 4 日

一、实验目的和要求

1. 熟悉通过 SQL 进行数据完整性控制的方法。

二、实验内容和要求

1. 建立表，考察表的生成者拥有该表的哪些权限。
2. 使用 SQL 的 grant 和 revoke 命令对其他用户进行授权和权力回收，考察相应的作用。
3. 建立视图，并把该视图的查询权限授予其他用户，考察通过视图进行权限控制的作用。
4. 完成实验报告。

三、主要仪器设备

1. 操作系统： Windows
2. 数据库管理系统： SQL Server 或 MySQL （本次实验选用 MySQL）

四、操作方法与实验步骤

4.1 考察表的生成者具有哪些权限

继续使用上次实验中我创建的数据库和表。也就是考察此时我所具有的权限。
首先查看用户列表：

```
select user,host from mysql.user;
```

找到其中的 root 用户，然后输入下列代码查看权限：

```
SELECT * FROM mysql.user WHERE user='root'\G
```

可以看到结果：我具有增删查改的全部权限。

4.2 建立超级用户 A

手动建立 public 和 owner 角色。创建数据库（登录）用户账号 A，以 public 和 owner 角色映射到 library 数据库上。

```
create role if not exists 'public', 'owner';
grant select on library.* to public;
grant insert, update, delete, select on library.* to owner with grant option;
create user A@localhost identified by 'A';
grant public to A@localhost;
grant owner to A@localhost;
```

这里建立了 public 和 owner 两种角色，给予了 public（所有人）查的权力，而把 owner 当作超级用户，他可以为所欲为。

查找此时的角色和用户列表，结果达到预期。

4.3 建立 public 的用户 B，检查他的角色权限

创建数据库（登录）用户账号 B，以 public 角色映射到 library 数据库上，以账户 B 登录，测试 B 能否对 book 表进行 CRUD（增、删、改、查）操作。

```
create user B@localhost identified by 'B';
grant public to B@localhost;
SELECT DISTINCT CONCAT('User: ''',user,'''@'',host,''';') AS query FROM mysql.user;
```

接下来我们用 B 用户进行登录，然后检查他的权限。

```
select * from book;
insert into book values('15', '数据库', 'MySQL', '浙江大学', 2003, '高鹏', 58.00, 24, 4);
delete from book where bno = '11';
update book set year = 2008 where cno = '11';
```

4.4 检查赋权后的效果变化

用 A 登录，利用 GRANT 语句赋予 B 表查询和插入的权限。

```
grant insert, select on library.* to B@localhost;
```

然后再次登录 B，测试 B 此时的权限：

```
select * from book;
insert into book values('15', '数据库', 'MySQL', '浙江大学', 2003, '高鹏', 58.00, 24, 4);
delete from book where bno = '11';
update book set year = 2008 where cno = '11';
```

4.5 收回权力，再检查权限

用 A 登录，利用 REVOKE 语句收回 book 表的操作权限，再进行测试。再用 B 登录，测试 B 的权限。

```
revoke insert, select on library.book from B@localhost;

insert into book values('16', '数学', '微积分', '浙江大学', 2013, '蔡明', 28.00, 54, 8);
delete from book where bno = '11';
update book set year = 2008 where cno = '11';
```

4.6 建立视图，并把该视图的查询权限授予其他用户，考察通过视图进行权限控制的作用。

首先建立视图：

```
create view book_view as select bno, title from book;
```

然后将此视图的增删查改权力赋给 B。

```
grant select,delete,insert,update on lab4_Library.book_view to B@localhost;
```

然后测试 B 对该视图的权力：

```
select * from book_view;
update book_view set author='test' where bno = '13';
update book_view set title='test' where bno = '13';
```

五、实验结果与分析

5.1 查看创建表的人的权限如下：

```
mysql> select user,host from mysql.user
-> ;
-> ||
```

user	host
mysql.infoschema	localhost
mysql.session	localhost
mysql.sys	localhost
root	localhost

```
4 rows in set (0.00 sec)

mysql> SELECT * FROM mysql.user WHERE user='root'\G;
***** 1. row *****
      Host: localhost
      User: root
  Select_priv: Y
  Insert_priv: Y
  Update_priv: Y
  Delete_priv: Y
  Create_priv: Y
```

可以看出此时我具有所有的权限。

5.2 建立超级用户 A 和 B 后的结果

```
mysql> SELECT DISTINCT CONCAT('User: ',user,'@',host,';') AS query FROM mysql.user;
-> ||
```

query
User: 'owner'@'%';
User: 'public'@'%';
User: 'A'@'localhost';
User: 'mysql.infoschema'@'localhost';
User: 'mysql.session'@'localhost';
User: 'mysql.sys'@'localhost';
User: 'root'@'localhost';

```
mysql> SELECT DISTINCT CONCAT('User: ',user,'@',host,';') AS query FROM mysql.user;
-> ||
```

query
User: 'owner'@'%';
User: 'public'@'%';
User: 'A'@'localhost';
User: 'B'@'localhost';
User: 'mysql.infoschema'@'localhost';
User: 'mysql.session'@'localhost';
User: 'mysql.sys'@'localhost';
User: 'root'@'localhost';

```
8 rows in set (0.00 sec)
```

可以看出，两次的结果均符合预期。

5.3 检查 B 具有的权限

```
mysql> use lab4_library
Database changed
mysql> select * from book;
```

bno	category	title	press	year	author	price	total	stock
11	生物学	物种起源	哈佛大学	2002	Darwin	48.00	20	4
12	计算机	深入理解计算机系统	浙江大学	2002	Darwin	48.00	20	4
13	计算机	算法导论	浙江大学	2002	Darwin	48.00	20	4
14	计算机	数据库系统原理	浙江大学	2002	Darwin	48.00	20	4

```
4 rows in set (0.22 sec)

mysql> insert into book values('15', '数据库', 'MySQL', '浙江大学', 2003, '高鹏', 58.00, 24, 4);
ERROR 1142 (42000): INSERT command denied to user 'B'@'localhost' for table 'book'
mysql> delete from book where bno = '11';
ERROR 1142 (42000): DELETE command denied to user 'B'@'localhost' for table 'book'
mysql> update book set year = 2008 where cno = '11';
ERROR 1142 (42000): UPDATE command denied to user 'B'@'localhost' for table 'book'
mysql>
```

可以看出，MySQL 拒绝了 B 的增删改操作，只允许查找。而我们对 public 的授权恰好也是这样。这和我们的预期是一致的。

5.4 检查 A 再次赋权后 B 的权限

用 A 给 B 赋权的结果：

```
mysql> grant insert, select on lab4_library.* to B@localhost;
Query OK, 0 rows affected (0.07 sec)
```

然后用 B 登陆后，检查权限如下：

```
mysql> use lab4_library
Database changed
mysql> select * from book;
```

bno	category	title	press	year	author	price	total	stock
11	生物学	物种起源	哈佛大学	2002	Darwin	48.00	20	4
12	计算机	深入理解计算机系统	浙江大学	2002	Darwin	48.00	20	4
13	计算机	算法导论	浙江大学	2002	Darwin	48.00	20	4
14	计算机	数据库系统原理	浙江大学	2002	Darwin	48.00	20	4

```
4 rows in set (0.01 sec)

mysql> insert into book values('15', '数据库', 'MySQL', '浙江大学', 2003, '高鹏', 58.00, 24, 4);
Query OK, 1 row affected (0.18 sec)

mysql> delete from book where bno = '11';
ERROR 1142 (42000): DELETE command denied to user 'B'@'localhost' for table 'book'
mysql> update book set year = 2008 where cno = '11';
ERROR 1142 (42000): UPDATE command denied to user 'B'@'localhost' for table 'book'
mysql>
```

说明此时 B 仍然只有查找和增加的权限。这又符合我们的预期。

5.5 收回权力后，再次查看 B 的权力

```
mysql> insert into book values('16', '数学', '微积分', '浙江大学', 2013, '蔡明', 28.00, 54, 8);
ERROR 1046 (3D000): No database selected
mysql> delete from book where bno = '11';
ERROR 1046 (3D000): No database selected
mysql> update book set year = 2008 where cno = '11';
ERROR 1046 (3D000): No database selected
mysql>
```

可以看出此时 B 什么权限都没有了。这符合我们的期望。

5.6 赋给 B 视图的权力，并检查效果

```
mysql> select * from book;
```

bno	category	title	press	year	author	price	total	stock
11	生物学	物种起源	哈佛大学	2002	Darwin	48.00	20	4
12	计算机	深入理解计算机系统	浙江大学	2002	Darwin	48.00	20	4
13	计算机	算法导论	浙江大学	2002	Darwin	48.00	20	4
14	计算机	数据库系统原理	浙江大学	2002	Darwin	48.00	20	4
15	数据库	MySQL	浙江大学	2003	高鹏	58.00	24	4

```
5 rows in set (0.02 sec)

mysql> create view book_view as select bno, title from book;
Query OK, 0 rows affected (0.76 sec)
```

此时我们创建了 bno 和 title 上的 book 的视图。

```
mysql> grant select,delete,insert,update on lab4_Library.book_view to B@localhost;
Query OK, 0 rows affected (0.25 sec)
```

这里我们将该视图的所有权限给了 B。

对 B 的权力测试的结果为：

```
mysql> use lab4_Library
Database changed
mysql> select * from book_view;
```

bno	title
11	物种起源
12	深入理解计算机系统
13	算法导论
14	数据库系统原理
15	MySQL

```
5 rows in set (0.04 sec)

mysql> update book_view set author='test' where bno = '13';
ERROR 1054 (42S22): Unknown column 'author' in 'field list'
mysql> update book_view set title='test' where bno = '13';
Query OK, 1 row affected (0.16 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> update book_view set title='test' where bno = '13';
Query OK, 1 row affected (0.16 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from book_view
-> ;
```

bno	title
11	物种起源
12	深入理解计算机系统
13	test
14	数据库系统原理
15	MySQL

```
5 rows in set (0.00 sec)
```

可以看出 B 对视图具有相应的功能。但是 B 的增加功能仅限于视图内，并不能通过给予 B 视图的权限从而让他有对于整个关系的操作权限。这两者是有很大的区别的。

六、讨论与心得

此次试验主要是考察数据库的角色用户以及权限的相关概念。

MySQL 没有 public 和 owner 的概念。

为了完成试验，我需要 MySQL 中自己创建 public 和 owner 的角色，并赋予了 public 查询的权限。在创建新用户后，如果未给用户赋予 public 角色，他依然不能查询，可知 MySQL 并不会给新角色自动赋予权限，需要管理员手动设置权限信息。

在收回权限的时候，用户 A 收回了用户 B 的增查权限，但是用户 B 由于具有 public 的角色，依然可以查询，所以以后涉及到撤销某个用户权限时，一定要仔细查看他是否还有其他用户或角色赋予的权限。