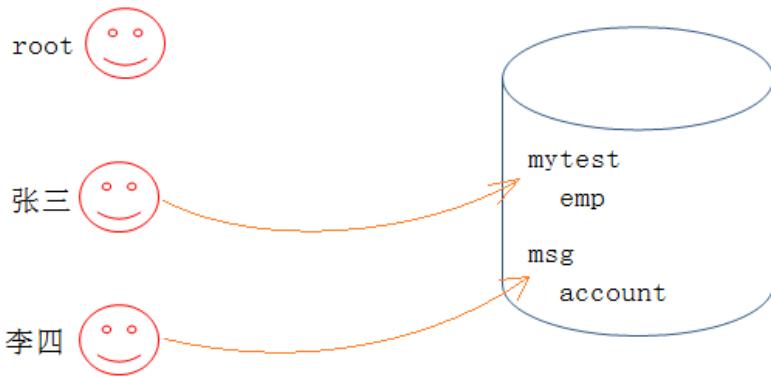


13. 用户管理

如果我们只能使用root用户，这样存在安全隐患。这时，就需要使用MySQL的用户管理。



张三只能操纵mytest这个库，李四只能操纵msg这个库。如果给他们root账户，那他们就可以操纵所有的库，风险太大。

13.1 用户

13.1.1 用户信息

MySQL中的用户，都存储在系统数据库mysql的user表中

```
mysql> use mysql;
Database changed
mysql> select host,user,authentication_string from user;
+-----+-----+-----+
| host | user | authentication_string |
+-----+-----+-----+
| localhost | root | *81F5E21E35407D884A6CD4A731AEBFB6AF209E1B |
| localhost | mysql.session | *THISISNOTAVALIDPASSWORDTHATCANBEUSEDHERE |
| localhost | mysql.sys | *THISISNOTAVALIDPASSWORDTHATCANBEUSEDHERE |
+-----+-----+-----+
```

--可以通过desc user初步查看一下表结构

字段解释：

- host: 表示这个用户可以从哪个主机登陆，如果是localhost，表示只能从本机登陆
- user: 用户名
- authentication_string: 用户密码通过password函数加密后的
- *_priv: 用户拥有的权限

13.1.2 创建用户

语法：

```
create user '用户名'@'登陆主机/ip' identified by '密码';
```

案例：

```
mysql> create user 'whb'@'localhost' identified by '12345678';
```

```

Query OK, 0 rows affected (0.06 sec)

mysql> select user,host,authentication_string from user;
+-----+-----+-----+
| user      | host     | authentication_string          |
+-----+-----+-----+
| root      | %        | *A2F7C9D334175DE9AF4DB4F5473E0BD0F5FA9E75 |
| mysql.session | localhost | *THISISNOTAVALIDPASSWORDTHATCANBEUSEDHERE |
| mysql.sys    | localhost | *THISISNOTAVALIDPASSWORDTHATCANBEUSEDHERE |
| whb        | localhost | *84AAC12F54AB666ECFC2A83C676908C8BBC381B1 | --新增用户
+-----+-----+-----+
4 rows in set (0.00 sec)

```

-- 此时便可以使用新账号新密码进行登陆啦

--备注：可能实际在设置密码的时候，因为mysql本身的认证等级比较高，一些简单的密码无法设置，会爆出如下报错：

```

-- ERROR 1819 (HY000): Your password does not satisfy the current policy
requirements
-- 解决方案: https://blog.csdn.net/zhanalou4821/article/details/93622812
-- 查看密码设置相关要求: SHOW VARIABLES LIKE 'validate_password%';
-- 这个大家下来自己玩玩

```

--关于新增用户这里，需要注意，不要轻易添加一个可以从任意地方登陆的user。

13.1.3 删除用户

语法：

```
drop user '用户名'@'主机名'
```

示例：

```

mysql> select user,host,authentication_string from user;
+-----+-----+-----+
| user      | host     | authentication_string          |
+-----+-----+-----+
| root      | %        | *A2F7C9D334175DE9AF4DB4F5473E0BD0F5FA9E75 |
| mysql.session | localhost | *THISISNOTAVALIDPASSWORDTHATCANBEUSEDHERE |
| mysql.sys    | localhost | *THISISNOTAVALIDPASSWORDTHATCANBEUSEDHERE |
| whb        | localhost | *84AAC12F54AB666ECFC2A83C676908C8BBC381B1 |
+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> drop user whb;           --尝试删除
ERROR 1396 (HY000): Operation DROP USER failed for 'whb'@'%'
-- <= 直接给个用户名，不能删除，它默认是%，表示所有地方可以登陆的用户

```

```

mysql> drop user 'whb'@'localhost'; --删除用户
Query OK, 0 rows affected (0.00 sec)

```

```

mysql> select user,host,authentication_string from user;
+-----+-----+-----+
| user      | host     | authentication_string          |
+-----+-----+-----+

```

```
+-----+-----+
| root | % | *A2F7C9D334175DE9AF4DB4F5473E0BD0F5FA9E75 |
| mysql.session | localhost | *THISISNOTAVALIDPASSWORDTHATCANBEUSEDHERE |
| mysql.sys | localhost | *THISISNOTAVALIDPASSWORDTHATCANBEUSEDHERE |
+-----+-----+
3 rows in set (0.00 sec)
```

13.1.4 修改用户密码

语法：

- 自己改自己密码

```
set password=password('新的密码');

--自己下来试试
```

- root用户修改指定用户的密码

```
set password for '用户名'@'主机名'=password('新的密码');
```

```
mysql> select host,user, authentication_string from user;
+-----+-----+-----+
| host | user | authentication_string |
+-----+-----+-----+
| % | root | *A2F7C9D334175DE9AF4DB4F5473E0BD0F5FA9E75 |
| localhost | mysql.session | *THISISNOTAVALIDPASSWORDTHATCANBEUSEDHERE |
| localhost | mysql.sys | *THISISNOTAVALIDPASSWORDTHATCANBEUSEDHERE |
| localhost | whb | *84AAC12F54AB666ECFC2A83C676908C8BBC381B1 |
+-----+-----+
4 rows in set (0.00 sec)
```

```
mysql> set password for 'whb'@'localhost'=password('87654321');
Query OK, 0 rows affected, 1 warning (0.00 sec)
```

```
mysql> select host,user, authentication_string from user;
+-----+-----+-----+
| host | user | authentication_string |
+-----+-----+-----+
| % | root | *A2F7C9D334175DE9AF4DB4F5473E0BD0F5FA9E75 |
| localhost | mysql.session | *THISISNOTAVALIDPASSWORDTHATCANBEUSEDHERE |
| localhost | mysql.sys | *THISISNOTAVALIDPASSWORDTHATCANBEUSEDHERE |
| localhost | whb | *5D24C4D94238E65A6407DFAB95AA4EA97CA2B199 |
+-----+-----+
4 rows in set (0.00 sec)
```

13.2 数据库的权限

MySQL数据库提供的权限列表：

权限	列	上下文
CREATE	Create_priv	数据库、表或索引
DROP	Drop_priv	数据库或表
GRANT OPTION	Grant_priv	数据库、表或保存的程序
REFERENCES	References_priv	数据库或表
ALTER	Alter_priv	表
DELETE	Delete_priv	表
INDEX	Index_priv	表
INSERT	Insert_priv	表
SELECT	Select_priv	表
UPDATE	Update_priv	表
CREATE VIEW	Create_view_priv	视图
SHOW VIEW	Show_view_priv	视图
ALTER ROUTINE	Alter_routine_priv	保存的程序
CREATE ROUTINE	Create_routine_priv	保存的程序
EXECUTE	Execute_priv	保存的程序
FILE	File_priv	服务器主机上的文件访问
CREATE TEMPORARY TABLES	Create_tmp_table_priv	服务器管理
LOCK TABLES	Lock_tables_priv	服务器管理
CREATE USER	Create_user_priv	服务器管理
PROCESS	Process_priv	服务器管理
RELOAD	Reload_priv	服务器管理
REPLICATION CLIENT	Repl_client_priv	服务器管理
REPLICATION SLAVE	Repl_slave_priv	服务器管理
SHOW DATABASES	Show_db_priv	服务器管理
SHUTDOWN	Shutdown_priv	服务器管理
SUPER	Super_priv	服务器管理

13.2.1 给用户授权

刚创建的用户没有任何权限。需要给用户授权。

语法:

```
grant 权限列表 on 库.对象名 to '用户名'@'登陆位置' [identified by '密码']
```

说明:

- 权限列表，多个权限用逗号分开

```
grant select on ...
```

```
grant select, delete, create on ....
```

```
grant all [privileges] on ... -- 表示赋予该用户在该对象上的所有权限
```

- `*.*`: 代表本系统中的所有数据库的所有对象（表，视图，存储过程等）
- `库.*`: 表示某个数据库中的所有数据对象(表，视图，存储过程等)
- `identified by`可选。如果用户存在，赋予权限的同时修改密码,如果该用户不存在，就是创建用户

案例：

```
--使用root账号  
--终端A  
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| 57test |  
| bit_index |  
| cadata_pro |  
| innodb_test |  
| musicserver |  
| myisam_test |  
| mysql |  
| order_sys |  
| performance_schema |  
| scott |  
| sys |  
| test |  
| vod_system |  
+-----+  
14 rows in set (0.00 sec)
```

```
mysql> use test;  
Database changed  
mysql> show tables;  
+-----+  
| Tables_in_test |  
+-----+  
| account |  
| student |  
| user |  
+-----+  
3 rows in set (0.01 sec)
```

```
--给用户whb赋予test数据库下所有文件的select权限  
mysql> grant select on test.* to 'whb'@'localhost';  
Query OK, 0 rows affected (0.01 sec)
```

```
--使用whb账号  
--终端B  
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
+-----+  
1 row in set (0.00 sec)
```

--暂停等root用户给whb赋完权之后，在查看

```
mysql> show databases;  
+-----+  
| Database |  
+-----+
```

```

| information_schema |
| test                | --赋完权之后，就能看到新的表
+-----+
2 rows in set (0.01 sec)

mysql> use test;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----+
| Tables_in_test |
+-----+
| account        |
| student         |
| user            |
+-----+
3 rows in set (0.00 sec)

mysql> select * from account;
+----+----+----+
| id | name | blance |
+----+----+----+
| 2  | 李四 | 321.00 |
| 3  | 王五 | 5432.00 |
| 4  | 赵六 | 543.90  |
| 5  | 赵六 | 543.90  |
+----+----+----+
4 rows in set (0.00 sec)

--没有删除权限
mysql> delete from account;
ERROR 1142 (42000): DELETE command denied to user 'whb'@'localhost' for table
'account'

```

备注：特定用户现有查看权限

```

mysql> show grants for 'whb'@'%';
+-----+
| Grants for whb@%                                |
+-----+
| GRANT USAGE ON *.* TO 'whb'@'%'                 |
| GRANT ALL PRIVILEGES ON `test`.* TO 'whb'@'%'   |
+-----+
2 rows in set (0.00 sec)

mysql> show grants for 'root'@'%';
+-----+
| Grants for root@%                               |
+-----+
| GRANT ALL PRIVILEGES ON *.* TO 'root'@'%' WITH GRANT OPTION |
+-----+
1 row in set (0.00 sec)

```

注意：如果发现赋权限后，没有生效，执行如下指令：

```
flush privileges;
```

13.2.2 回收权限

语法：

```
revoke 权限列表 on 库.对象名 from '用户名'@'登陆位置';
```

示例：

```
-- 回收whb对test数据库的所有权限  
--root身份, 终端A  
mysql> revoke all on test.* from 'whb'@'localhost';  
Query OK, 0 rows affected (0.00 sec)  
  
--whb身份, 终端B  
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| test |  
+-----+  
2 rows in set (0.00 sec)  
  
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
+-----+  
1 row in set (0.00 sec)
```