



2021

数据库

SHUJUKU

微信公众号：灰灰考研



数据库高级管理

- 了解数据备份的基本概念
- 掌握MySQL中数据备份、数据恢复的基本操作
- 掌握MySQL中创建、删除用户的基本操作
- 掌握MySQL中授予、查看权限和收回权限的基本操作

微信公众号：灰灰考研

一、数据库的备份与恢复



数据备份是对数据库结构、对象和数据的复制，以便在数据库遭到破坏或因需求改变而需要把数据还原到某个时间点时能够恢复数据库。数据恢复就是将数据库备份加载到系统中。数据备份和数据恢复可以保护数据库的关键数据。

1. 数据备份的分类

(1) 按备份时服务器是否在线来划分。

①热备份。数据库正处于运行状态，此时依赖数据库的日志文件进行备份。

②温备份。进行数据备份时数据库服务正常运行，但数据只能读不能写。

③冷备份。数据库处于关闭状态，能够较好地保证数据库的完整性。

(2) 按备份的内容来划分。

①逻辑备份。使用软件从数据库中提取数据并将结果写到一个文件上，该文件格式一般与原数据库的文件格式不同，只是原数据库中数据内容的一个映像。

②物理备份。物理备份指直接复制数据库文件。与逻辑备份相比，其速度较快，但占用空间较大。

(3) 按备份涉及的数据范围来划分。

①完整备份。完整备份指备份整个数据库。这是任何备份策略中都要求完成的第一种备份类型，其他所有备份类型都依赖于完整备份。换句话说，如果没有执行完整备份，就无法执行差异备份和增量备份。

②增量备份。是对数据库从上一次完整备份或者最近一次增量备份以来改变的内容的备份。

③差异备份。是对最近一次完整备份以后发生改变的数据进行的备份。差异备份仅备份自最近一次完整备份后发生更改的数据。

备份是一种十分耗费时间和资源的操作，对其使用不能太过频繁。应该根据数据库使用情况确定一个合适的备份周期。

2. 数据恢复的方法数据恢复就是指当数据库出现故障时，将备份的数据库加载到系统中，从而使数据库恢复到备份时的正确状态。MySQL有三种保证数据安全的方法。

(1) 数据库备份。通过导出数据或者表文件的副本来保护数据。

(2) 二进制日志文件保存更新数据的所有语句。

(3) 数据库复制。MySQL的内部复制功能建立在两个或多个服务器之间，是通过设定它们之间的主从关系来实现的。其中一个作为主服务器，其他的作为从服务器。

一 (1) 数据库的备份

数据库的备份为了保证数据的安全，数据库管理员应定期对数据库进行备份。

备份需要遵循两个简单原则：

一是要尽早并且经常备份；

二是不要只备份到同一磁盘的同一文件中，要在不同位置保存多个副本，以确保备份安全。

MySQL的mysqldump命令可以实现数据的备份：

mysqldump命令可以备份单个数据库、多个数据库和所有数据库，由于这三种备份方式比较类似，所以本节就以备份单个数据库为例来讲解mysqldump命令，其他方式只列举语法格式。

1. 备份单个数据库mysqldump命令备份数据库的语法格式如下：

```
mysqldump -u username -h host -p password  
dbname [tbname1 . tbname2...] > filename.sql;
```

DVS

VR

2. 备份多个数据库使用mysqldump命令备份多个数据库时，需要使用--databases参数，其语法格式如下：

```
mysqldump -u username -h host -ppassword --database dbname1 [dbname2 dbname3...]>
filename.sql
```

3. 备份所有数据库使用mysqldump命令备份所有数据库时，需在该命令后使用--all-databases参数，其语法格式如下：

```
mysqldump -uusername -hhost -ppassword --all-databases>filename.sql
```

mysqldump命令提供了许多参数，包括用于调试和压缩的参数。

下面只列举最常用的参数：

运行帮助命令mysqldump--help，可以获得特定版本的完整参数列表。

--all-databases：备份所有数据库。

--databases dbname：备份某个数据库。

--lock-tables：锁定表。

--lock-all-tables：锁定所有的表。

--events：备份EVENT的相关信息。

--no-data：只备份DDL语句和表结构，不备份数据。

--master-data=n：备份的同时导出二进制日志文件及其位置。如果n为1，则把信息保存为CHANGE MASTER语句；如果n为2，则把信息保存为注释掉的CHANGEMASTER语句。

--routines：将存储过程和存储函数定义备份。

--single-transaction：实现热备份。

--triggers：备份触发器。

一 (2) 数据库的恢复

恢复数据库，就是当数据库中的数据遭到破坏时，让数据库根据备份的数据回到备份时的状态。这里所说的恢复是指恢复数据库中的数据，而数据库是不能被恢复的。通过前面的内容可知，备份文件实际上就是由多个CREATE、INSERT和DROP语句组成的，因此只要使用mysql命令执行这些语句就可以将数据恢复。

对于使用mysqldump命令备份后形成的.sql文件，可以使用mysql命令导入到数据库中。备份的.sql文件中包含CREATE、INSERT语句，也可能包含DROP语句。mysql命令可以直接执行文件中的这些语句。

mysql命令恢复数据的语法格式如下：

```
mysql -u username -ppassword [dbname] <filename.sql
```


二、
用户管
用户分
二 (1)
在安装
tables_
信息以

字段名称	字段含义	默认值
Host	主机名	
User	用户名	
Password	密码	
Select_priv	确定用户是否可以通过 SELECT 命令选择数据	N
Insert_priv	确定用户是否可以通过 INSERT 命令插入数据	N
Update_priv	确定用户是否可以通过 UPDATE 命令修改现有数据	N
Delete_priv	确定用户是否可以通过 DELETE 命令删除现有数据	N
Create_priv	确定用户是否可以创建新的数据库和表	N
Drop_priv	确定用户是否可以删除现有数据库和表	N
Reload_priv	确定用户是否可以执行用来刷新和重新加载 MySQL 所用的各种内部缓存（包括日志、权限、主机、查询和表）的特定命令	N
Shutdown_priv	确定用户是否可以关闭 MySQL 服务器。在将此权限提供给 root 账户之外的任何用户时，都应当非常谨慎	N
ssl_type	用于加密	
ssl_cipher	用于加密	NULL
max_questions	每小时允许用户执行查询操作的次数	0
max_updates	每小时允许用户执行更新操作的次数	0
max_connections	每小时允许用户建立连接的次数	0
max_user_connections	允许单个用户同时建立连接的次数	0

， MySQL
旨定权限。
db、 host、
器的账号
|字段如表：

二 (2) 创建新用户

在MySQL数据库中，只有一个root用户是无法管理众多数据的，因此需要创建多个普通用户来管理不同类型的数据。创建普通用户有以下三种方法。1. 使用CREATE USER语句创建新用户
使用CREATE USER语句创建一个新用户时，MySQL服务器会自动修改相应的授权表。该语句创建的新用户可以连接数据库，但是不具有任何权限。CREATE USER语句创建用户的语法格式如下：

```
CREATE USER 'username' @ 'hostname' [IDENTIFIED BY [PASSWORD] 'password'  
[, 'username' @ 'hostname' [IDENTIFIED BY [PASSWORD] 'password' ] ]...
```

2. 使用GRANT语句创建用户与CREATE USER语句不同，GRANT语句不仅可以创建新用户，而且可以对用户授权。除此之外，GRANT语句还可以指定用户的其他特点，如安全连接、限制使用服务器资源等。GRANT语句执行的时候自动加载权限表，无需手动刷新。因此，GRANT语句是创建用户最常用的方法。GRANT语句创建用户的语法格式如下：

```
GRANT privileges ON database.table TO 'username' '@' hostname' [IDENTIFIED BY  
[PASSWORD] 'password' ] [, 'username' '@' hostname' [IDENTIFIED BY [PASSWORD]  
D] 'password' ]...
```

3. 使用INSERT语句创建用户从上面的讲述中可以看出，不管是使用CREATE USER语句还是GRANT语句创建新用户，实际上就是在mysql数据库的user表中添加一条新的记录。因此，可以使用INSERT语句直接将新用户的信息添加到mysql.user表中。

使用INSERT语句创建用户的语法格式如下：

```
INSERT INTO mysql.user (Host, User, Password, ssl_cipher, x509_issue  
r, x509_subject )  
VALUES ('hostname', 'username', PASSWORD ('password'), '', '', '')
```

二 (3) 删除普通用户

1. 使用DROP USER语句删除用户如果要删除某个用户，只需在DROP USER后面指定要删除的用户即可。

DROPUSER语句还可以同时删除多个MySQL账户，并取消其权限。要使用DROP USER语句，必须拥有mysql数据库的DROP USER权限。

DROP USER语句的语法格式如下：

```
DROP USER 'username' @'hostname' [, 'username' @'hostname'];
```

2. 使用DELETE语句删除用户所谓删除用户，就是将mysql数据库的user表中的一条记录删除，因此，也可以通过DELETE语句实现用户删除。使用DELETE语句删除user表中的数据时，要指定表名为mysql.user，用户还必须拥有对mysql.user表的DELETE权限。

DELETE语句的语法格式如下：

```
DELETE FROM mysql.user WHERE Host='hostname' AND User='username' ;
```

二（4）修改用户信息

创建好用户后，还可以对用户的名称和密码进行修改！

1. 修改用户名称在MySQL中，修改用户的名称可以使用RENAME USER语句。

基本语法格式如下：

```
RENAME USER old_username TO new_username[, old_username TO new_username] [, ...]
```

2. 修改用户密码

MySQL中的用户可以对数据库进行权限内的各种操作，因此用户的密码非常重要，密码一旦泄露就必须马上修改。在MySQL用户中，root用户的权限最高，可以修改自己的密码，也可以修改其他普通用户的密码；普通用户只能修改自己的密码。

修改某个用户的登录密码，可以使用mysqladmin命令、UPDATE语句或SETPASSWORD语句来实现。

由于~~root用户和普通用户~~修改密码的方式比较类似，下面就以修改root用户的密码为例进行讲解。

(1) 使用mysqladmin命令修改密码mysqladmin命令通常用于执行一些管理性的工作，在MySQL中可以使用该命令来修改用户的密码。

mysqladmin命令修改密码的语法格式如下：

```
mysqladmin -u username [-h hostname] -p password new_password
```

(2) 使用UPDATE语句修改root用户密码修改用户密码信息，也可以直接修改mysql数据库的user表。root用户只要登录到MySQL服务器，就可以使用UPDATE语句直接修改自己的密码。UPDATE语句修改密码的语法格式如下：

```
UPDATE mysql.user set Password= PASSWORD('new_password') WHERE User='username' and Host='hostname';
```


(3) 使用SET语句修改root用户的密码要想使用SET语句修改密码，首先使用root用户登录服务器。新密码一定要使用PASSWORD () 函数加密。SET语句修改密码的语法格式如下：

```
SET PASSWORD [FOR username]=PASSWORD ('newpassword');
```

三、权限管理

在MySQL数据库中，为了保证数据的安全性，数据库管理员需要为每个用户赋予不同的权限，以满足不同用户的需求。对权限管理简单的理解就是MySQL控制用户只能做权限以内的事情，不可以越界。

权限管理主要是对登录到MySQL的用户进行权限验证。所有用户的权限都存储在MySQL的权限表中。合理的权限管理能够保证数据库系统的安全，而不合理的权限设置会给MySQL服务器带来安全隐患。

三 (1) M
权限管理:
权限管理:

续表

权限名称	权限范围	权限说明
INDEX	表	添加索引的权限
INSERT	表	插入数据的权限
SELECT	表	查询数据的权限
UPDATE	表	更新数据的权限
CREATE VIEW	视图	创建视图的权限
SHOW VIEW	视图	查看视图的权限
ALTER ROUTINE	存储过程、函数	更改存储过程或函数的权限
CREATE ROUTINE	存储过程、函数	创建存储过程或函数的权限
EXECUTE	存储过程、函数	执行存储过程或函数的权限
FILE	服务器主机上的文件	文件访问权限
CREATE TEMPORARY TABLES	表	创建临时表权限
LOCK TABLES	表	锁表权限
CREATE USER	服务器管理	创建用户权限
PROCESS	服务器管理	查看进程权限
RELOAD	服务器管理	执行 flush hosts、flush logs、flush privileges、flush status、flush tables、flush threads、refresh、reload 等命令的权限
REPLICATION CLIENT	服务器管理	复制权限
REPLICATION SLAVE	服务器管理	复制权限
SHOW DATABASES	服务器管理	查看数据库权限
SHUTDOWN	服务器管理	关闭数据库权限
SUPER	服务器管理	执行 kill 线程权限

合理的

三（2）权限查询

MySQL提供SHOW GRANTS语句用来显示指定用户的权限信息。SHOW GRANTS的基本语法格式如下：

```
SHOW GRANTS FOR 'username'@'hostname';
```

三（3）权限授予

在MySQL中，可以使用GRANT语句为用户授予权限。GRANT语句的语法格式如下：

```
GRANT priv_type [(columns)][, priv_type[(columns)]] ON database.table  
TO 'username'@'hostname' [IDENTIFIED BY [PASSWORD]'password']  
[, 'username'@  
hostname' [IDENTIFIED BY [PASSWORD]'password']]...[WITH with_option  
[with_option]...]
```

三（4）权限收回

在MySQL中，为了保证数据的安全，有时需要取消已经授予用户的某些权限。权限收回后，用户账户记录将从db、host、tables_priv和columns_priv表中删除，但是用户账户记录仍然在user表中保存。收回权限利用REVOKE语句来实现。

1. 收回指定权限REVOKE语句的语法格式如下：

```
REVOKE priv_type [columns] [, priv_type [(columns)]] ON databas  
e.table  
FROM 'username' :
```

2. 收回所有权限收回所有权限的语法格式如下：

```
REVOKE ALL PRIVILEGES, GRANT OPTION FROM 'username' @'hostname'  
[, 'username'  
@'hostname' ]...
```

本章小结

- 数据的备份与恢复。为保证数据库中数据的安全，要经常进行数据库的备份。当数据库遭到破坏时，可以使用备份来恢复数据。
- 数据库用户的管理，包括数据库用户的增加、删除、修改、查看等基本操作。
- 用户权限的管理，包括用户权限的查询、授予和收回等操作。

课后习题

1. 什么是数据库的备份和恢复？为什么要进行数据库的备份？
2. 创建用户的方法有哪几种？写出这几种方法。
3. 写出为用户授予权限的命令。
4. 如果root用户的密码泄露，请问该怎么解决？



THANKS

感谢观看

微信公众号：灰灰考研

