

# Linux 系统下 mysql 的备份与还原

## 一【实验目标】

- 学习并掌握 Linux 下 MySQL 的备份与还原

## 二【实验环境】

- 实验机环境：Centos 6.6
- 目标机环境：Centos 6.6
- 实验拓扑：如图 1 所示。



图 1 实验拓扑

## 三【实验原理】

在 Linux 下，为了防止数据库数据的丢失，需要备份来以防万一。

## 四【实验步骤】

- 1、先用命令 `sudo netstat -tap|grep mysql` 检测该 Linux 下是否有 mysql 的存在。输入命令后没有反应，证明不存在，再使用命令 `sudo apt-get install mysql-server mysql-client` 安装 mysql，如下图。

```
yb524555@ubuntu:~$ sudo netstat -tap|grep mysql
[sudo] password for yb524555:
yb524555@ubuntu:~$ sudo apt-get install mysql-server mysql-client
```

图 1

- 2、安装成功之后，再使用相同命令检测 MySQL 是否存在。如图显示，该 Linux 下已经存在了 MySQL。使用命令 `mysql -uroot -p` 登录

```
yb524555@ubuntu:~$ sudo netstat -tap|grep mysql
tcp        0      0 localhost:mysql    *:*          LISTEN
7308/mysql
yb524555@ubuntu:~$ mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 42
Server version: 5.5.47-0ubuntu0.14.04.1 (Ubuntu)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
```

图 2

- 3、增加一个新的 database，名字为 yb，然后往 yb 里简单的加入一个 table yb1，再往里加入一行数据。

```
at line 1
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| yb |
+-----+
4 rows in set (0.00 sec)

mysql> use yb;
Database changed
mysql> show yb;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that
corresponds to your MySQL server version for the right syntax to use near 'yb' a
at line 1
mysql> select *from yb1;
Empty set (0.02 sec)

mysql> insert into yb1 values(78,'Andy');
Query OK, 1 row affected (0.01 sec)
```

图 3

```
mysql> select *from yb1;
+----+-----+
| id | name |
+----+-----+
| 78 | Andy |
+----+-----+
1 row in set (0.00 sec)
```

图 4

#### 4、mysql 的备份

先用 `su root` 语句以管理员的方式运行 terminal  
然后进入 `cd /var/lib/MySQL` 目录下

再执行语句 `mysqldump -u root -p yb>1.sql`  
将数据库 yb 里的东西备份到 1.sql 中

```
yb524555@ubuntu:~$ su root
Password:
root@ubuntu:/home/yb524555# cd/
bash: cd/: No such file or directory
root@ubuntu:/home/yb524555# cd /
root@ubuntu:/# cd /var/lib/mysql
root@ubuntu:/var/lib/mysql# mysqldump -u root -p yb>1.sql
Enter password:
root@ubuntu:/var/lib/mysql# mysqldump -u root -p yb>2.sql
Enter password:
mysqldump: Got error: 1045: Access denied for user 'root'@'localhost' (using password: YES) when trying to connect
root@ubuntu:/var/lib/mysql# vi 1.sql

[1]+  Stopped                  vi 1.sql
root@ubuntu:/var/lib/mysql# cd /
root@ubuntu:/# cd /var/lib/mysql
root@ubuntu:/var/lib/mysql# vi 1.sql
```

图 5

使用命令 `vi 1.sql` 查看 1.sql 中的内容

```
-- MySQL dump 10.13  Distrib 5.5.47, for debian-linux-gnu (x86_64)
--
-- Host: localhost    Database: yb
--
-- Server version      5.5.47-0ubuntu0.14.04.1

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

--
-- Table structure for table `yb1`
--

DROP TABLE IF EXISTS `yb1`;
```

图 6

```

/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `yb1` (
  `id` int(11) NOT NULL,
  `name` char(20) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `yb1`
--

LOCK TABLES `yb1` WRITE;
/*!40000 ALTER TABLE `yb1` DISABLE KEYS */;
INSERT INTO `yb1` VALUES (78,'Andy');
/*!40000 ALTER TABLE `yb1` ENABLE KEYS */;
UNLOCK TABLES;
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;

```

图 7

```

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

-- Dump completed on 2016-03-21 11:42:27

```

图 8

## 5、MySQL 的还原

还原之前建立一个新的 database lbj。

```

mysql> create database
-> ;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that
corresponds to your MySQL server version for the right syntax to use near 'dateb
ase' at line 1
mysql> create database lbj;
Query OK, 1 row affected (0.01 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| lbj |
| mysql |
| performance_schema |
| yb |
+-----+

```

图 9

往该数据库中还原备份过的东西，语句 `mysql -u root -p lbj<1.sql`

```

root@ubuntu:/# cd /var/lib/mysql
root@ubuntu:/var/lib/mysql# mysql -u root -p lbj<1.sql

```

图 10

查询 lbj 中的 table yb1 中的数据

```
mysql> select *from yb1;
+-----+-----+
| id | name |
+-----+-----+
| 78 | Andy |
+-----+-----+
1 row in set (0.01 sec)
```

图 11

这说明备份过的 1.sql 中的内容已经还原到了 lbj 中。

## 五【实验思考】

- 有关 MySQL 文件夹的访问权限问题。这从另一个层面说明了 MySQL 数据库本身就对数据有一定的保护力度。