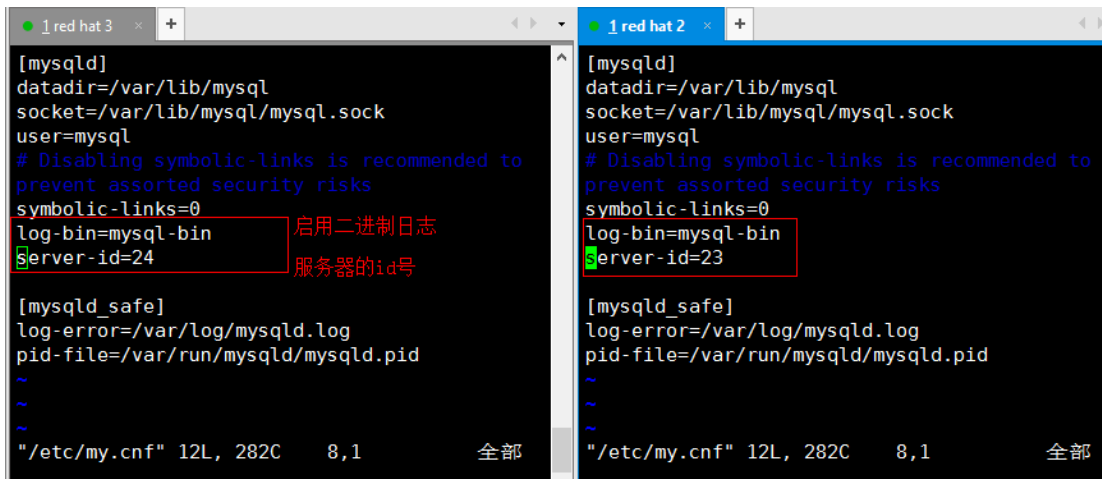


MySQL 集群

1. MySQL 主从备份

前提条件：安装了 mysql, 开启了二进制日志



```
[mysql]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to
prevent assorted security risks
symbolic-links=0
log-bin=mysql-bin
server-id=24

[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid

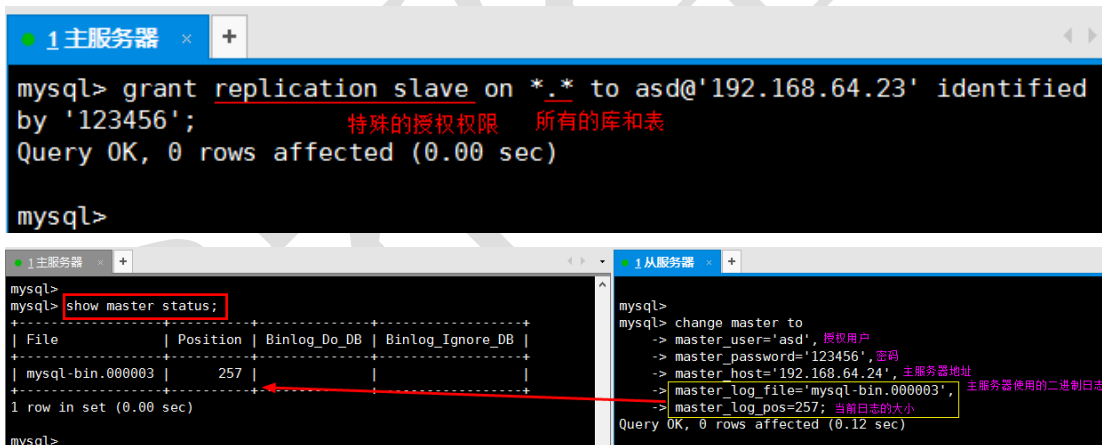
"/etc/my.cnf" 12L, 282C 8,1 全部
```

```
[mysql]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to
prevent assorted security risks
symbolic-links=0
log-bin=mysql-bin
server-id=23

[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid

"/etc/my.cnf" 12L, 282C 8,1 全部
```

在**主服务器上授权**，从服务器保存授权的信息



```
mysql> grant replication slave on *.* to asd@'192.168.64.23' identified
by '123456';
Query OK, 0 rows affected (0.00 sec)

mysql>
```

```
mysql> show master status;
+-----+-----+-----+-----+
| File           | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+-----+-----+-----+-----+
| mysql-bin.000003 | 257      |               |                   |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

```
mysql> change master to
-> master_user='asd',
-> master_password='123456',
-> master_host='192.168.64.24',
-> master_log_file='mysql-bin.000003',
-> master_log_pos=257;
```

之后在从服务器会产生授权信息文件

```
mysql> exit
Bye
[root@localhost mysql]# ls
aa          mysql-bin.000001      mysqld-relay-bin.index
ibdata1     mysql-bin.000002      mysql.sock
ib_logfile0 mysql-bin.000003      relay-log.info
ib_logfile1 mysql-bin.index        test
master.info  mysqld-relay-bin.000001
mysql        mysqld-relay-bin.000002
[root@localhost mysql]# cat master.info
15
mysql-bin.000003
336
192.168.64.24
asd
123456
3306
60
0
0
0
[root@localhost mysql]#
```

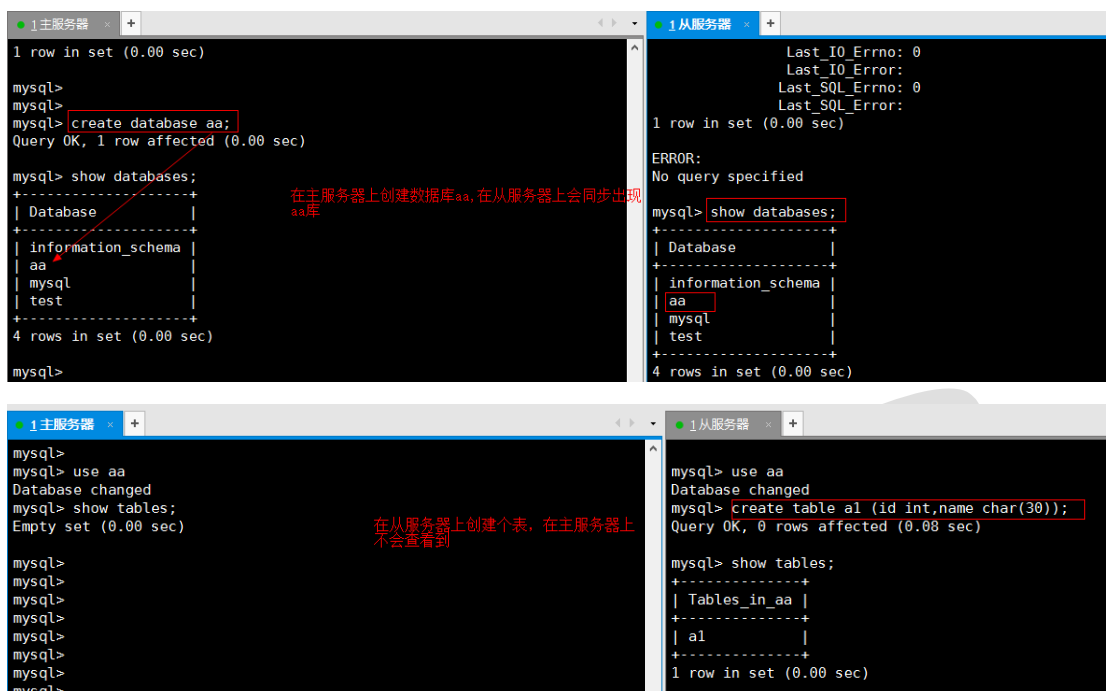
授权信息内容

开启从服务器 start slave, 并查看

```
mysql> start slave; 开启从服务器
Query OK, 0 rows affected (0.01 sec)

mysql> show slave status\G; 查看从服务器的内容
***** 1. row *****
Slave_IO_State: Waiting for master to send event
Master_Host: 192.168.64.24
Master_User: asd
Master_Port: 3306
Connect_Retry: 60
Master_Log_File: mysql-bin.000003
Read_Master_Log_Pos: 257
Relay_Log_File: mysqld-relay-bin.000002
Relay_Log_Pos: 251
Relay_Master_Log_File: mysql-bin.000003
Slave_IO_Running: Yes
Slave_SQL_Running: Yes 线程已启用
Replicate_Do_DB:
Replicate_Ignore_DB:
Replicate_Do_Table:
```

测试



The screenshots show the setup of MySQL replication. The first screenshot shows the master server (1主服务器) where a database 'aa' is created. The second screenshot shows the slave server (1从服务器) where the same database 'aa' is created. The third screenshot shows the master server where a table 'a1' is created. The fourth screenshot shows the slave server where the same table 'a1' is created. Red annotations highlight the database and table names across the different servers.

```

1主服务器
mysql> create database aa;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| aa        |
| mysql     |
| test      |
+-----+
4 rows in set (0.00 sec)

1从服务器
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| aa        |
| mysql     |
| test      |
+-----+
4 rows in set (0.00 sec)

1主服务器
mysql> use aa;
Database changed
mysql> show tables;
Empty set (0.00 sec)

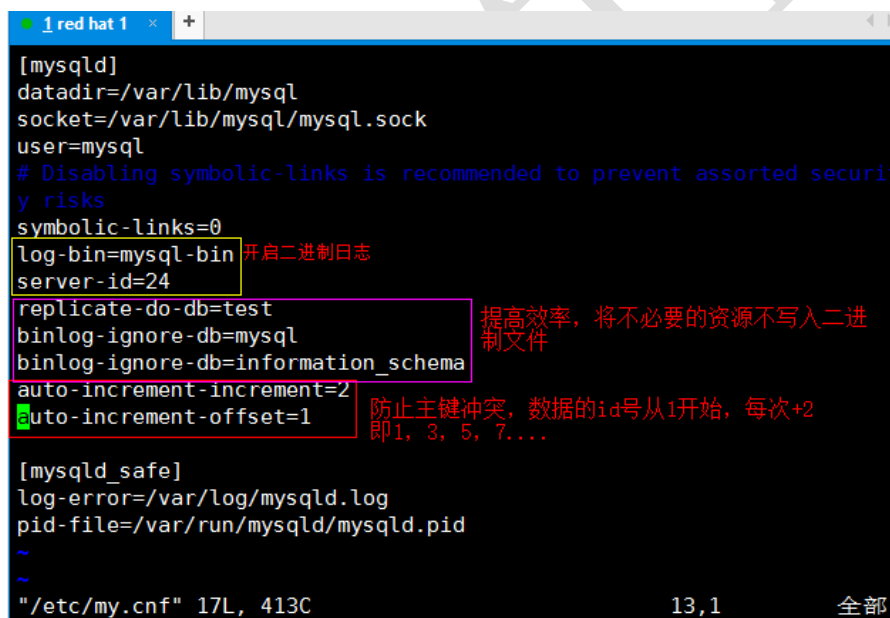
1从服务器
mysql> use aa;
Database changed
mysql> create table a1 (id int,name char(30));
Query OK, 0 rows affected (0.00 sec)

mysql> show tables;
+-----+
| Tables_in_aa |
+-----+
| a1            |
+-----+
1 row in set (0.00 sec)

```

2. MySQL 主主备份

1. 以 1 为主，2 为从配置一遍主从
在主配置文件中配置一下（开启二进制日志和其他内容）



The screenshot shows the MySQL configuration file (/etc/my.cnf) with various settings for the master server. Red annotations highlight specific settings and their purposes.

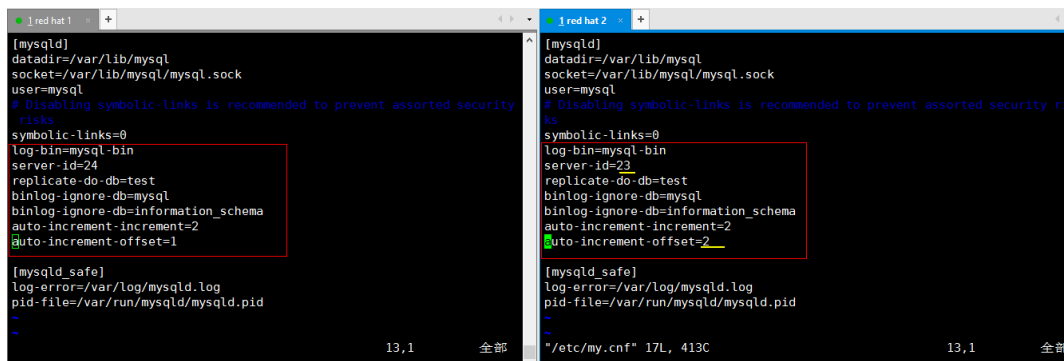
```

[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to prevent assorted security risks
symbolic-links=0
log-bin=mysql-bin 开启二进制日志
server-id=24
replicate-do-db=test
binlog-ignore-db=mysql
binlog-ignore-db=information_schema
auto-increment-increment=2
auto-increment-offset=1 防止主键冲突，数据的id号从1开始，每次+2即1, 3, 5, 7....

[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid
~
"/etc/my.cnf" 17L, 413C

```

2. 在 2 上做相同的配置



3. 启动服务器

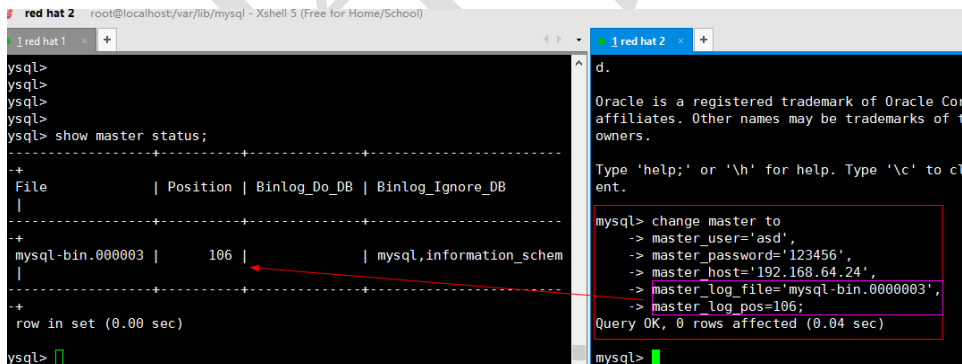
```
[root@localhost mysql]# vim /etc/my.cnf
[root@localhost ~]# service mysqld start
初始化 MySQL 数据库: Installing MySQL system tables...
OK
Filling help tables...
OK
```

1 为主 2 为从:

在主服务器 (1) 上授权

```
mysql> grant replication slave on *.* to asd@'192.168.64.23' identified by '123456';
Query OK, 0 rows affected (0.00 sec)
```

在从服务器 (2) 上保存授权信息



2 为主 1 为从:

在主服务器 (2) 上授权

```

-> master_log_pos=106;
Query OK, 0 rows affected (0.04 sec)

mysql> grant replication slave on *.* to asd@'192.168.64.24' identified by
'123456';
Query OK, 0 rows affected (0.00 sec)

mysql>
mysql>

```

在从服务器(1)上保存授权信息

```

mysql>
mysql>
mysql>
mysql> change master to
-> master_user='asd',
-> master_password='123456',
-> master_host='192.168.64.23',
-> master_log_file='mysql-bin.000001',
-> master_log_pos=106;
Query OK, 0 rows affected (0.05 sec)

mysql>
mysql>

```

```

'123456';
Query OK, 0 rows affected (0.00 sec)

mysql>
mysql> show master status;
+-----+-----+-----+-----+
| File | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+-----+-----+-----+-----+
| mysql-bin.000001 | 106 | | mysql,information_schem |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

```

1 和 2 都执行 start slave (互为主从)

```

6', master_host='192.168.64.23', master_log_file='mysql-bin.00000
1', master_log_pos=106;
Query OK, 0 rows affected (0.33 sec)

mysql> start slave;
Query OK, 0 rows affected (0.01 sec)

mysql> show slave status\G;
***** 1. row *****
Slave_IO_State: Waiting for master to send event
Master_Host: 192.168.64.23
Master_User: asd
Master_Port: 3306
Connect_Retry: 60
Master_Log_File: mysql-bin.000001
Read_Master_Log_Pos: 199
Relay_Log_File: mysqld-relay-bin.000002
Relay_Log_Pos: 335
Relay_Master_Log_File: mysql-bin.000001
Slave_IO_Running: Yes
Slave_SQL_Running: Yes
Replicate_Do_DB: test

```

```

mysql> change master to master_user='asd', master_password='123456', master_hos
t='192.168.64.24', master_log_file='mysql-bin.000003', master_log_pos=106;
Query OK, 0 rows affected (0.04 sec)

mysql> start slave;
Query OK, 0 rows affected (0.00 sec)

mysql> show slave status\G;
***** 1. row *****
Slave_IO_State: Waiting for master to send event
Master_Host: 192.168.64.24
Master_User: asd
Master_Port: 3306
Connect_Retry: 60
Master_Log_File: mysql-bin.000003
Read_Master_Log_Pos: 190
Relay_Log_File: mysqld-relay-bin.000002
Relay_Log_Pos: 335
Relay_Master_Log_File: mysql-bin.000003
Slave_IO_Running: Yes
Slave_SQL_Running: Yes
Replicate_Do_DB: test

```

测试

```

mysql> use test;
Database changed
mysql> create table a1(id int);
Query OK, 0 rows affected (0.09 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| bb |
| mysql |
| test |
+-----+
4 rows in set (0.00 sec)

mysql> show tables;
+-----+
| Tables_in_test |
+-----+
| a1 |
+-----+
1 row in set (0.00 sec)

mysql> show tables;
+-----+
| Tables_in_test |
+-----+
| a1 |
| a2 |
+-----+
2 rows in set (0.00 sec)

```

```

ERROR:
No query specified

mysql> show tables;
ERROR 1046 (3D000): No database selected
mysql> use test;
Reading table information for completion of table and
You can turn off this feature to get a quicker startup

Database changed
mysql> show tables;
+-----+
| Tables_in_test |
+-----+
| a1 |
+-----+
1 row in set (0.00 sec)

mysql> create table a2;
ERROR 1113 (42000): A table must have at least 1 column
mysql> create table a2(id int);
Query OK, 0 rows affected (0.03 sec)

mysql> show tables;
+-----+
| Tables_in_test |
+-----+
| a1 |
| a2 |
+-----+
2 rows in set (0.00 sec)

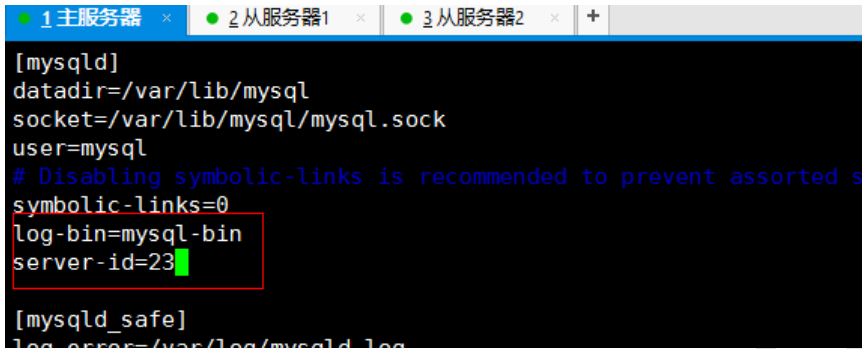
mysql>

```

3. MySQL 一主多从

主服务器配置

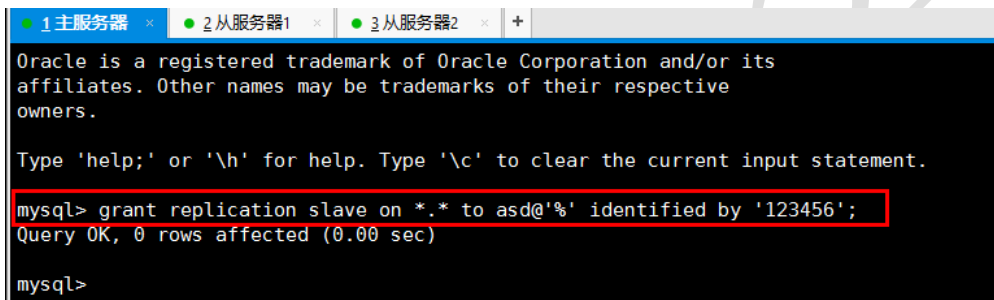
开启二进制日志，并启动 mysql



```
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to prevent assorted s
symbolic-links=0
log-bin=mysql-bin
server-id=23

[mysqld_safe]
log-error=/var/log/mysqld.log
```

在主服务器上授权



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

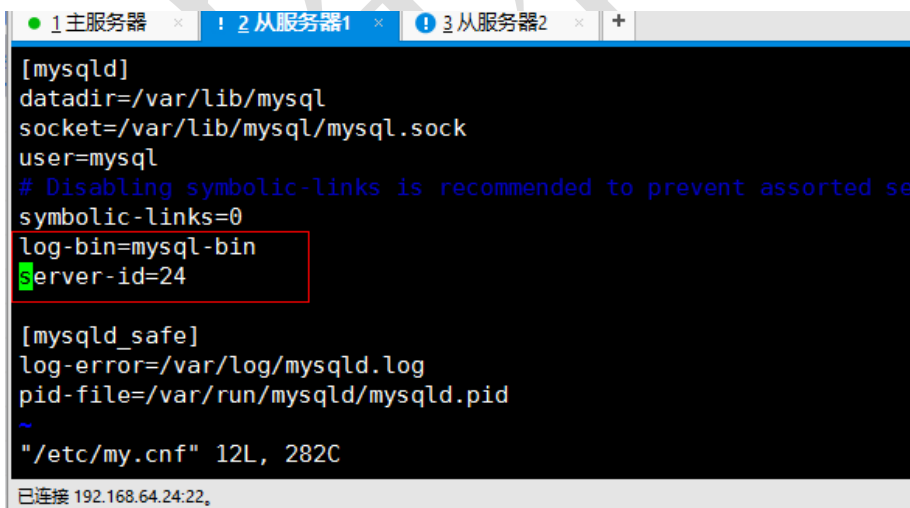
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> grant replication slave on *.* to asd@'%' identified by '123456';
Query OK, 0 rows affected (0.00 sec)

mysql>
```

从服务器配置

开启二进制日志，并启动 mysql



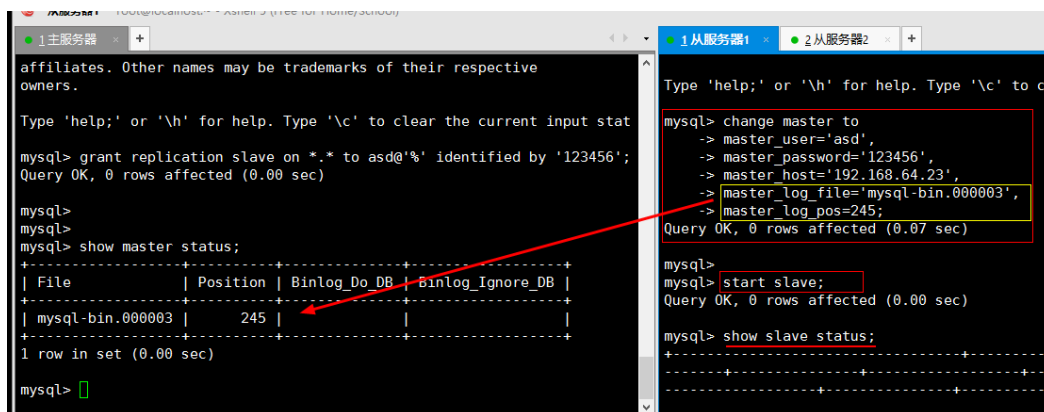
```
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to prevent assorted se
symbolic-links=0
log-bin=mysql-bin
server-id=24

[mysqld_safe]
log-error=/var/log/mysqld.log
pid-file=/var/run/mysqld/mysqld.pid

"/etc/my.cnf" 12L, 282C

已连接 192.168.64.24:22.
```

保存授权信息



```

1 主服务器
affiliates. Other names may be trademarks of their respective owners.

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

mysql> grant replication slave on *.* to asd@'%' identified by '123456';
Query OK, 0 rows affected (0.00 sec)

mysql>
mysql>
mysql> show master status;
+-----+-----+-----+-----+
| File           | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+-----+-----+-----+-----+
| mysql-bin.000003 | 245      |               |                   |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>

1 从服务器1
Type 'help;' or '\h' for help. Type '\c' to clear the current input stat

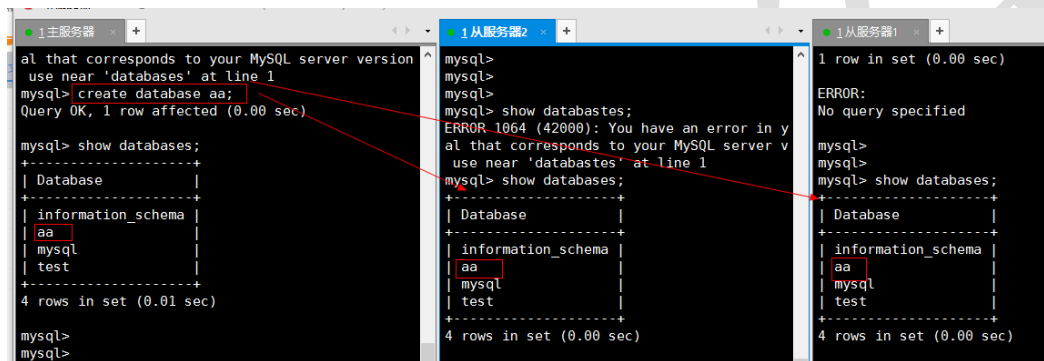
mysql> change master to
-> master_user='asd',
-> master_password='123456',
-> master_host='192.168.64.23',
-> master_log_file='mysql-bin.000003',
-> master_log_pos=245;
Query OK, 0 rows affected (0.07 sec)

mysql>
mysql> start slave;
Query OK, 0 rows affected (0.00 sec)

mysql> show slave status;
+-----+-----+-----+-----+
| File           | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+-----+-----+-----+-----+
| mysql-bin.000003 | 245      |               |                   |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
    
```

#在另一台从服务器上做相同的配置（注意 id 不能相同）

测试



```

1 主服务器
al that corresponds to your MySQL server version
use near 'databases' at line 1
mysql> create database aa;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| aa        |
| mysql     |
| test      |
+-----+
4 rows in set (0.01 sec)

mysql>
mysql>

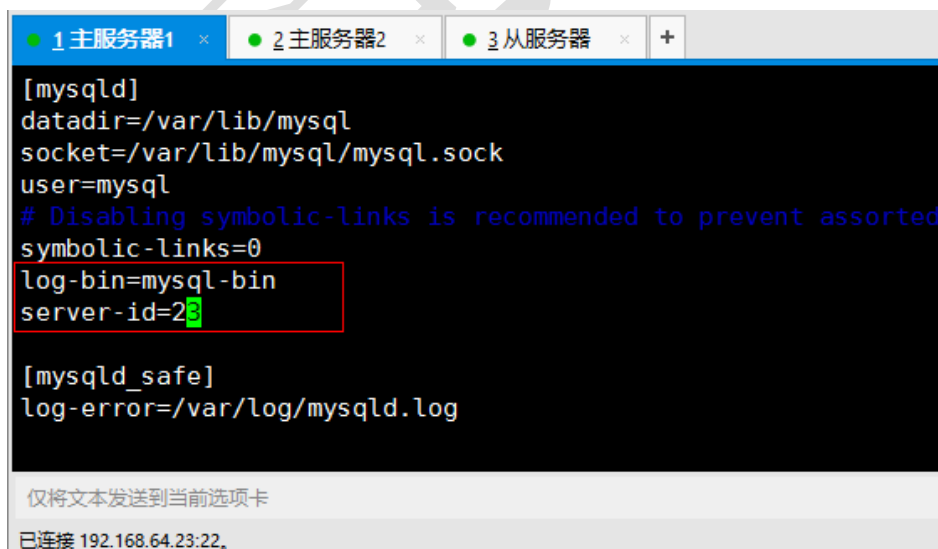
1 从服务器2
mysql>
mysql>
mysql> show databases;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version 5.7.26 near 'databases' at line 1
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| aa        |
| mysql     |
| test      |
+-----+
4 rows in set (0.00 sec)

1 从服务器1
1 row in set (0.00 sec)
ERROR:
No query specified
mysql>
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| aa        |
| mysql     |
| test      |
+-----+
4 rows in set (0.00 sec)
    
```

4. MySQL 多主一从

主服务器配置

开启二进制日志，启动服务



```

1 主服务器1
2 主服务器2
3 从服务器

[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to prevent assorted
symbolic-links=0
log-bin=mysql-bin
server-id=2

[mysqld_safe]
log-error=/var/log/mysqld.log

仅将文本发送到当前选项卡
已连接 192.168.64.23:22.
    
```

授权

```
mysql> grant replication slave on *.* to asd@'192.168.64.24' identified by '123456';
Query OK, 0 rows affected (0.00 sec)

mysql>
```

在主服务器 2 上做相同的操作

```
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to prevent assorted security
symbolic-links=0
log-bin=mysql-bin
server-id=25

[mysqld_safe]
log-error=/var/log/mysqld.log
```

授权

```
mysql> grant replication slave on *.* to asd@'192.168.64.24' identified by '123456';
Query OK, 0 rows affected (0.00 sec)
```

从服务器操作
对主配置文件操作


```
[mysqld]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to prevent assorted security risks
symbolic-links=0

[mysqld_safe]
log-error=/var/log/mysql.log
pid-file=/var/run/mysqld/mysqld.pid

##M-M-S

[mysqld_multi]
mysqld=/usr/bin/mysqld_safe
mysqladmin=/usr/bin/mysqladmin
log=/tmp/multi.log

[mysqld10]
port=3306
datadir=/var/lib/mysql/
pid-file=/var/lib/mysql/mysqld.pid
socket=/var/lib/mysql/mysql.sock
user=mysql
server-id=30

[mysqld20]
port=3307
datadir=/var/lib/mysql/
pid-file=/var/lib/mysql/mysqld.pid
socket=/var/lib/mysql/mysql.sock
user=mysql
server-id=30
~
```

初始化数据库，生成目录 mysqla, mysqlb

```
1 主服务器1 x 2 主服务器2 x 3 从服务器 x +
[root@localhost ~]# vim /etc/my.cnf
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# mysql_install_db --datadir=/var/lib/mysqla --user=mysql
Installing MySQL system tables... 初始化数据库，生成mysqla
OK
Filling help tables...
OK
```

```
1 主服务器1 x 2 主服务器2 x 3 从服务器 x +
Please report any problems with the /usr/bin/mysqlbug script!

[root@localhost ~]# mysql_install_db --datadir=/var/lib/mysqlb --user=mysql
Installing MySQL system tables... 生成mysqlb
OK
Filling help tables...
OK

To start mysqld at boot time you have to copy
support-files/mysql.server to the right place for your system
```

```

1 主服务器1  2 主服务器2  3 从服务器
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# cd /var/lib/
[root@localhost lib]# ls
alternatives  certmonger  fprint      logrotate.status  mysql      nfs      polkit-1  random-seed
authconfig    dbus        games       misc              mysqla     ntp      postfix   readahead
cas           dhcclient   ipa-client  mlocate          mysqlb     plymouth prelink    rhsm
[root@localhost lib]# ls -lh mysqla/
总用量 8.0K
drwx----- 2 mysql root 4.0K 2月 25 07:26 mysql
drwx----- 2 mysql root 4.0K 2月 25 07:26 test
[root@localhost lib]# ls -lh mysqlb/
总用量 8.0K
drwx----- 2 mysql root 4.0K 2月 25 07:31 mysql
drwx----- 2 mysql root 4.0K 2月 25 07:31 test
[root@localhost lib]#

```

设置 mysqla, mysqlb 目录及以下文件的属主为 mysql (防止出现权限问题)

```

[root@localhost lib]# chown -R mysql /var/lib/mysqla/
[root@localhost lib]# chown -R mysql /var/lib/mysqlb/
[root@localhost lib]#
[root@localhost lib]#

```

启动从服务器线程

```

[root@localhost ~]# mysqld_multi --defaults-file=/etc/my.cnf start 23
[root@localhost ~]# netstat -anpt
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN      1160/rpcbind
tcp        0      0 0.0.0.0:57680           0.0.0.0:*               LISTEN      1178/rpc.statd
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      1382/sshd
tcp        0      0 127.0.0.1:631            0.0.0.0:*               LISTEN      1259/cupsd
tcp        0      0 127.0.0.1:25             0.0.0.0:*               LISTEN      1458/master
tcp        0      0 0.0.0.0:3306             0.0.0.0:*               LISTEN      2107/mysqld
tcp        0      0 192.168.64.24:22         192.168.64.1:58561     ESTABLISHED 1717/sshd
tcp        0      0 192.168.64.24:22         192.168.64.1:50013     ESTABLISHED 1646/sshd
tcp        0      0 :::53903                 :::*                     LISTEN      1178/rpc.statd
tcp        0      0 :::111                   :::*                     LISTEN      1160/rpcbind
tcp        0      0 :::22                    :::*                     LISTEN      1382/sshd
tcp        0      0 :::1:631                 :::*                     LISTEN      1259/cupsd
tcp        0      0 :::1:25                   :::*                     LISTEN      1458/master

[root@localhost ~]# mysqld_multi --defaults-file=/etc/my.cnf start 25
[root@localhost ~]# netstat -anpt
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:3307             0.0.0.0:*               LISTEN      5245/mysqld
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN      1160/rpcbind
tcp        0      0 0.0.0.0:57680           0.0.0.0:*               LISTEN      1178/rpc.statd
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      1382/sshd
tcp        0      0 127.0.0.1:631            0.0.0.0:*               LISTEN      1259/cupsd
tcp        0      0 127.0.0.1:25             0.0.0.0:*               LISTEN      1458/master
tcp        0      0 0.0.0.0:3306             0.0.0.0:*               LISTEN      2107/mysqld
tcp        0      264 192.168.64.24:22         192.168.64.1:58561     ESTABLISHED 1717/sshd
tcp        0      0 192.168.64.24:22         192.168.64.1:50013     ESTABLISHED 1646/sshd
tcp        0      0 :::53903                 :::*                     LISTEN      1178/rpc.statd
tcp        0      0 :::111                   :::*                     LISTEN      1160/rpcbind
tcp        0      0 :::22                    :::*                     LISTEN      1382/sshd
tcp        0      0 :::1:631                 :::*                     LISTEN      1259/cupsd
tcp        0      0 :::1:25                   :::*                     LISTEN      1458/master

```

登录并保存授权信息

```
● 1主服务器1 × ● 2主服务器2 × ● 3从服务器 × +
[root@localhost ~]# mysql -P 3306 -S /var/lib/mysql/mysql.sock
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1
Server version: 5.1.66 Source distribution

Copyright (c) 2000, 2012, 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> change master to
-> master_user='asd',
-> master_password='123456',
-> master_host='192.168.64.23',
-> master_log_file='mysql-bin.000003',
-> master_log_pos=257;
Query OK, 0 rows affected (0.09 sec)

mysql> start slave;
Query OK, 0 rows affected (0.00 sec)

mysql> show slave status\G;
***** 1. row *****
Slave_IO_State: Waiting for master to send event
Master_Host: 192.168.64.23
Master_User: asd
Master_Port: 3306
Connect_Retry: 60
```

```
● 1主服务器1 × ● 2主服务器2 × ● 3从服务器 × +

mysql>
mysql>
mysql> exit
Bye
[root@localhost ~]# mysql -P 3307 -S /var/lib/mysqlb/mysql.sock
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1
Server version: 5.1.66 Source distribution

Copyright (c) 2000, 2012, 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.

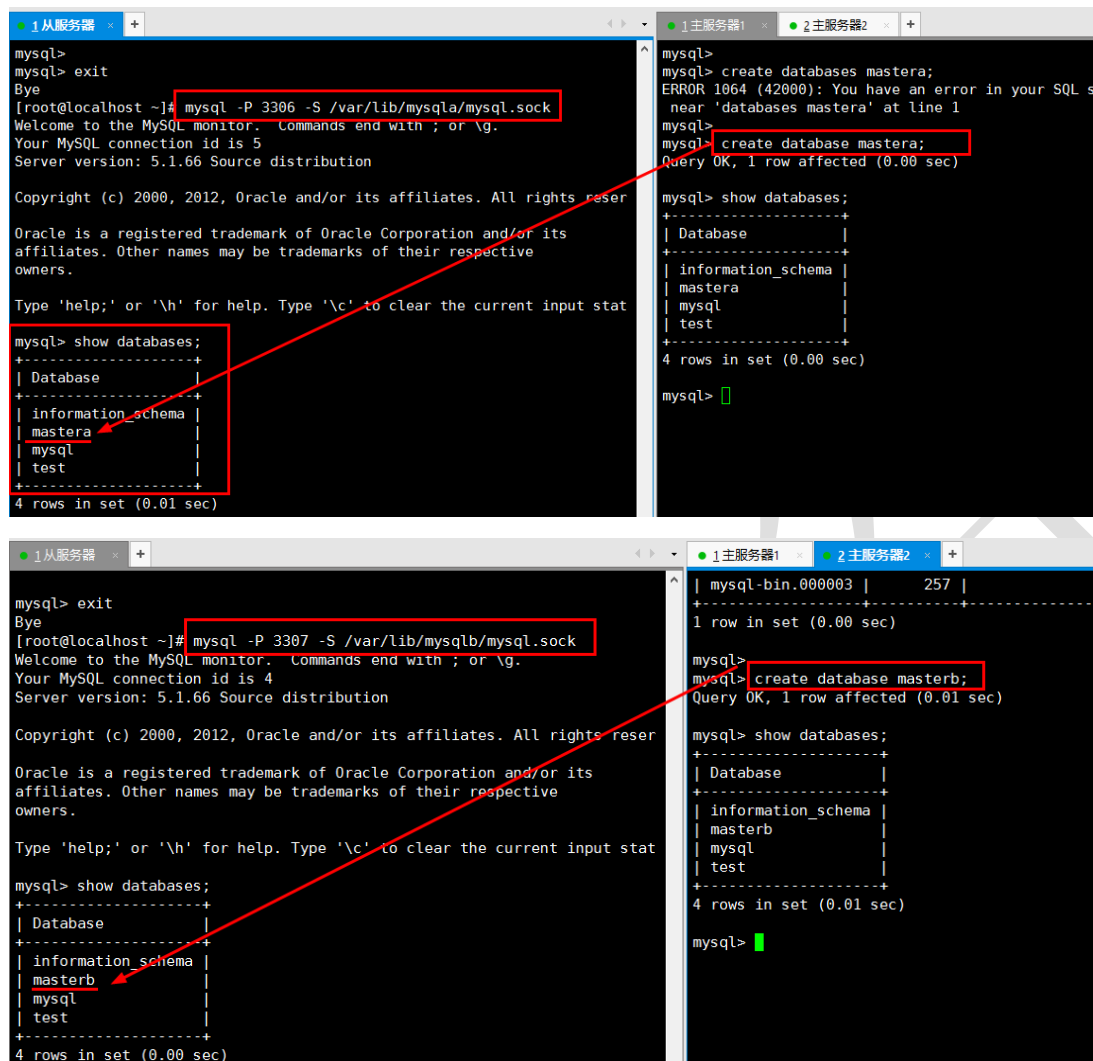
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> change master to
-> master_user='asd',
-> master_password='123456',
-> master_host='192.168.64.25',
-> master_log_file='mysql-bin.000003',
-> master_log_pos=257;
Query OK, 0 rows affected (0.10 sec)

mysql> start slave;
Query OK, 0 rows affected (0.00 sec)

mysql> show slave status;
```

测试



The screenshots show the following commands and outputs:

- Terminal 1 (从服务器):**

```
mysql> exit
Bye
[root@localhost ~]# mysql -P 3306 -S /var/lib/mysql/mysql.sock
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.1.66 Source distribution

Copyright (c) 2000, 2012, 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;
+-----+
| Database |
+-----+
| information_schema |
| mastera |
| mysql |
| test |
+-----+
4 rows in set (0.01 sec)
```
- Terminal 2 (主服务器1):**

```
mysql> create databases mastera;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that
near 'databases mastera' at line 1

mysql> create database mastera;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mastera |
| mysql |
| test |
+-----+
4 rows in set (0.00 sec)
```
- Terminal 3 (从服务器):**

```
mysql> exit
Bye
[root@localhost ~]# mysql -P 3307 -S /var/lib/mysqlb/mysql.sock
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 4
Server version: 5.1.66 Source distribution

Copyright (c) 2000, 2012, 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;
+-----+
| Database |
+-----+
| information_schema |
| masterb |
| mysql |
| test |
+-----+
4 rows in set (0.00 sec)
```
- Terminal 4 (主服务器2):**

```
mysql-bin.000003 | 257 |
+-----+
| Database |
+-----+
| information_schema |
| masterb |
| mysql |
| test |
+-----+
4 rows in set (0.01 sec)

mysql> create database masterb;
Query OK, 1 row affected (0.01 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| masterb |
| mysql |
| test |
+-----+
4 rows in set (0.01 sec)
```

5. MySQL 中间件-Amoeba

中间件：一种提供在不同技术、不同的软件之间共享资源的程序，更大化了利用了数据库的性能，可以无限扩展（注：真实环境中并非如此）

数据库的中间件：

- mysql proxy（官方版本）性能低，需要 lua 脚本
- atlas 性能低，响应时间长
- amoeba 陈思儒研发的

一．先搭建一个主从关系的服务器

在主、从服务器上安装 mysql mysql-server

```

mount: block device /dev/sr0 is write-protected, mounting read-only
[root@localhost ~]# yum -y install mysql mysql-server
Loaded plugins: product-id, security, subscription-manager
This system is not registered to Red Hat Subscription Management. You can use
subscription-manager to register.
rhel-source | 3.9 kB
rhel-source/primary.db | 3.1 MB
Setting up Install Process
Resolving Dependencies
--> Running transaction check
--> Package mysql.x86_64 0:5.1.66-2.el6_3 will be installed

[c:\>]#
Connecting to 192.168.64.25:22...
Connection established.
To escape to local shell, press 'Ctrl+Alt+J'.

Last login: Mon Mar 13 23:05:08 2017
[root@localhost ~]# mount /dev/sr0 /mnt/
mount: block device /dev/sr0 is write-protected, mounting read-only
[root@localhost ~]# vim /etc/yum.repos.d/rhel-source.repo
[root@localhost ~]#
  
```

1. 开启二进制日志

```

[mysql]
datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
user=mysql
# Disabling symbolic-links is recommended to prevent assorted security risks
symbolic-links=0
log-bin=mysql-bin
server-id=24

"/etc/my.cnf" 12L, 282C      8,1      顶端
  
```

2. 在主服务器上授权，从服务器上保存授权信息，并开启从服务线程。

```

mysql> change master to
-> master_user='asd',
-> master_password='123456',
-> master_host='192.168.64.23',
-> master_log_file='mysql-bin.000003',
-> master_log_pos=245;
Query OK, 0 rows affected (0.08 sec)

mysql> start slave;
Query OK, 0 rows affected (0.00 sec)

mysql> show slave status\G;

mysql> grant replication slave on *.* to asd@'%' identified by '123456';
Query OK, 0 rows affected (0.00 sec)

mysql> show masters;
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 'maste
rs' at line 1

mysql> show master status;
+-----+-----+-----+-----+
| File | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+-----+-----+-----+-----+
| mysql-bin.000003 | 245 | | |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
  
```

3. 关闭从服务器线程，为了做读写分离时，测试有明显的实验效果（实际生产环境中不能停掉。。）

```

1从(读) x +
mysql> stop slave;
Query OK, 0 rows affected (0.00 sec)

mysql> show slave status\G;
***** 1. row *****
      Slave_IO_State:
            Master_Host: 192.168.64.23
            Master_User: asd
            Master_Port: 3306
            Connect_Retry: 60
            Master_Log_File: mysql-bin.000003
            Read_Master_Log_Pos: 324
            Relay_Log_File: mysqld-relay-bin.000002
            Relay_Log_Pos: 330
            Relay_Master_Log_File: mysql-bin.000003
            Slave_IO_Running: No
            Slave_SQL_Running: No
            Replicate_Do_DB:

```

二. 配置读写分离

1. 安装 gcc 环境 (amoeba 需要源码安装)

```

[root@localhost ~]#
[root@localhost ~]# yum -y install gcc*
Loaded plugins: product-id, security, subscription-manager
This system is not registered to Red Hat Subscription Management. You can use subscri
rhel-source
rhel-source/primary_db
Setting up Install Process
Resolving Dependencies
--> Running transaction check
--> Package gcc.x86_64 0:4.4.7-3.el6 will be installed
--> Processing Dependency: cpp = 4.4.7-3.el6 for package: gcc-4.4.7-3.el6.x86_64

```

2. 拷贝第三方软件, 创建单独的目录

```

[root@localhost ~]# mount /dev/sr0 /mnt/
mount: block device /dev/sr0 is write-protected, mounting read-only
[root@localhost ~]# cd /mnt/
[root@localhost mnt]# ls
amoeba-mysql-1.3.1-BETA.zip  jdk-7u40-linux-x64.gz
[root@localhost mnt]# cp * /usr/src/
[root@localhost mnt]# cd !$
cd /usr/src/
[root@localhost src]# ls
amoeba-mysql-1.3.1-BETA.zip  debug  jdk-7u40-linux-x64.gz  kernels

[root@localhost src]# cd
[root@localhost ~]# mkdir /amoeba
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]#

```

创建单独的目录, 用来存储安装 amoeba 相关的东西

3. 先安装 jdk (amoeba 是由 java 语言编写的, 所以先安装 jdk), 配置 java 环境

```
[root@localhost src]# tar -xf jdk-7u40-linux-x64.gz -C /amoeba/
[root@localhost src]#
[root@localhost src]# cd /amoeba/
[root@localhost amoeba]# ls
jdk1.7.0_40
[root@localhost amoeba]# ln -s jdk1.7.0_40/ jdk
[root@localhost amoeba]# ls
jdk jdk1.7.0_40
[root@localhost amoeba]#
```

解压到指定的目录下

创个软链接去除版本号，方便操作

4. 声明用 java 写出来的程序如何调用 (/etc/profile)

```
# /etc/profile

# System wide environment and startup programs, for login setup
# Functions and aliases go in /etc/bashrc

# It's NOT a good idea to change this file unless you know what you
# are doing. It's much better to create a custom.sh shell script in
# /etc/profile.d/ to make custom changes to your environment, as this
# will prevent the need for merging in future updates.
JAVA_HOME=/amoeba/jdk
export JAVA_HOME

PATH=$JAVA_HOME/bin:$PATH
export PATH

CLASSPATH=.:$JAVA_HOME/bin/tools.jar:$JAVA_HOME/lib/dt.jar:$CLASSPATH
export CLASSPATH
```

```
[root@localhost amoeba]#
[root@localhost amoeba]# source /etc/profile
[root@localhost amoeba]#
[root@localhost amoeba]#
```

加载一下，使之生效

```
[root@localhost amoeba]#
[root@localhost amoeba]# java -version
java version "1.7.0_40"
Java(TM) SE Runtime Environment (build 1.7.0_40-b43)
Java HotSpot(TM) 64-Bit Server VM (build 24.0-b56, mixed mode)
[root@localhost amoeba]#
```

测试一下

5. 安装 amoeba

```
[root@localhost amoeba]# cd /usr/src/
[root@localhost src]# ls
amoeba-mysql-1.3.1-BETA.zip debug jdk-7u40-linux-x64.gz kernels
[root@localhost src]#
[root@localhost src]#
[root@localhost src]# unzip amoeba-mysql-1.3.1-BETA.zip -d /usr/local/amoeba
Archive: amoeba-mysql-1.3.1-BETA.zip
  creating: /usr/local/amoeba/bin/
  inflating: /usr/local/amoeba/bin/amoeba
  inflating: /usr/local/amoeba/bin/amoeba.bat
  creating: /usr/local/amoeba/doc/
  creating: /usr/local/amoeba/lib/
  creating: /usr/local/amoeba/lib/classes/
  inflating: /usr/local/amoeba/LICENSE.txt

[root@localhost src]#
[root@localhost src]# chmod -R +x /usr/local/amoeba/bin/
[root@localhost src]#
[root@localhost src]#
```

解压到指定位置下，用 -d

为了amoeba程序下的命令正常执行，给予权限并不是非做不可的操作

配置 amoeba 这个软件

```
[root@localhost src]# cd /usr/local/amoeba/
[root@localhost amoeba]# ls
bin  build.properties  build.xml  conf  doc  lib  LICENSE.txt  README.html  src
[root@localhost amoeba]# cd conf/
[root@localhost conf]# ls
access_list.conf  amoeba.dtd  amoeba.xml  function.dtd  functionMap.xml  log4j.dtd  log4j.xml  rule.dtd  ruleFunctionMap.xml  rule.xml
[root@localhost conf]#
```

存放了配置文件
主配置文件

<server>.....</server>区域

```
7      <!-- proxy server 的 IP 地址 -->
8      <property name="port">8066</property> 端口
9  server区域的配置
10     <!-- proxy server 的 IP 地址 -->
11
12     <property name="ipAddress">192.168.64.25</property> amoeba 安装的 IP 地址
13
14     <!-- proxy server net IO Read thread size -->
15     <property name="readThreadPoolSize">20</property>
16
17     <!-- proxy server client process thread size --> 使用默认值
18     <property name="clientSideThreadPoolSize">30</property>
19
20     <!-- mysql server data packet process thread size -->
21     <property name="serverSideThreadPoolSize">30</property>
22
23     <!-- socket Send and receive BufferSize(unit:K) -->
24     <property name="netBufferSize">128</property>
25
26     <!-- Enable/disable TCP_NODELAY (disable/enable Nagle's algorithm). -->
27     <property name="tcpNoDelay">true</property>
28
29     <!-- 用户名 -->
30     <property name="user">asd</property> 登录 amoeba 的用户名，本来是 root (和管理员用户没关系)
31
32     <!-- 密码 -->
33
34     <property name="password">123456</property> 登录用户使用的密码
35
36
37     <!-- query timeout( default: 60 second , TimeUnit:second) -->
38     <property name="queryTimeout">60</property>
39
40
41 <connectionManagerList>
42 连接时管理组件的类名
43 <connectionManager name="defaultManager" class="com.meidusa.amoeba.net.MultiConnectionManagerWrapper">
44     <property name="subManagerClassName">com.meidusa.amoeba.net.AuthingableConnectionManager</property>
45
46     <!-- default value is available Processors -->
47     <property name="processors">5</property> 不需要管
48     -->
49 </connectionManager>
```

<dbServerList>.....</dbServerList>区域

由于只提供了一个服务器模板，需要自己复制另一个填写关于读的

17

```
<dbServer name="slave" virtual="true">
  <poolConfig class="com.meidusa.amoeba.server.MultipleServerPool">
    <!-- 1=ROUNDROBIN, 2=WEIGHTBASED, 3=HA -->
    <property name="loadbalance">1</property>

    <!-- 1=IOE, 2=pool, 3=âpâpoolNameâ ± i 00q o o Â 0, i -->
    <property name="poolNames">server1,server2</property>
  </poolConfig>
</dbServer>
```

读从server1 server2

```
181 <property name="LRUMapSize">1500</property>
182 <property name="defaultPool">master</property> 默认访问池
183
184
185 <property name="writePool">master</property> 写入池
186 <property name="readPool">slave</property> 读取池
187
188 <property name="needParse">true</property>
189 </queryRouter>
190 </amoeba:configuration>
-- 插入 --
```

启动 amoeba, 修改一下启动脚本: /usr/local/amoeba/bin/amoeba

```
DEFAULT_OPTS="-server -Xms256m -Xmx256m -Xss256k"
# DEFAULT_OPTS="$DEFAULT_OPTS -XX:+HeapDumpOnOutOfMemoryError -XX:+AggressiveOpts -XX:+UseParallelGC -Xmx64m"
DEFAULT_OPTS="$DEFAULT_OPTS -Damoeba.home=\"$AMOEB_HOME\""
DEFAULT_OPTS="$DEFAULT_OPTS -Dclassworlds.conf=\"$AMOEB_HOME/bin/amoeba.classworlds\""

CMD="exec \"\$JAVA_HOME/bin/java\" \$DEFAULT_OPTS \$OPTS -classpath \"\$CLASSPATH\" \$MAIN_CLASS \$@"
eval $CMD
```

注意: 将-Xss128k 修改为 -Xss256

在主机和服务器上指定用户授权, 授权目的为了让 amoeba 能连接到主从服务器进行查询。

nohup bash -x /usr/local/amoeba/bin/amoeba &

把这个放到后台 退出终端也可以继续运行

```
[root@localhost bin]# nohup bash -x /usr/local/amoeba/bin/amoeba &
[1] 2635
[root@localhost bin]# nohup: 忽略输入并把输出追加到"nohup.out"
```

ps aux | grep amoeba

查看一下运行的程序 查看到的话就说明程序已经运行了起来

```
[root@localhost bin]# ps aux | grep amoeba
root      2635 17.4  5.5 1072680 56092 pts/0    Sl   00:05   0:03 /amoeba/jdk/bin/java -s
moeba -Dclassworlds.conf=/usr/local/amoeba/bin/amoeba.classworlds -classpath /usr/local/a
cher
root      2663  0.0  0.0 103240   872 pts/0    S+   00:06   0:00 grep amoeba
[root@localhost bin]#
```

测试 (安装一个 MySQL 软件包才可以连接)

```
[root@localhost conf]# mysql -uasd -p -h 192.168.64.25 -P 8066
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1290243769
Server version: 5.1.45-mysql-amoeba-proxy-1.3.1-BETA Source distribution

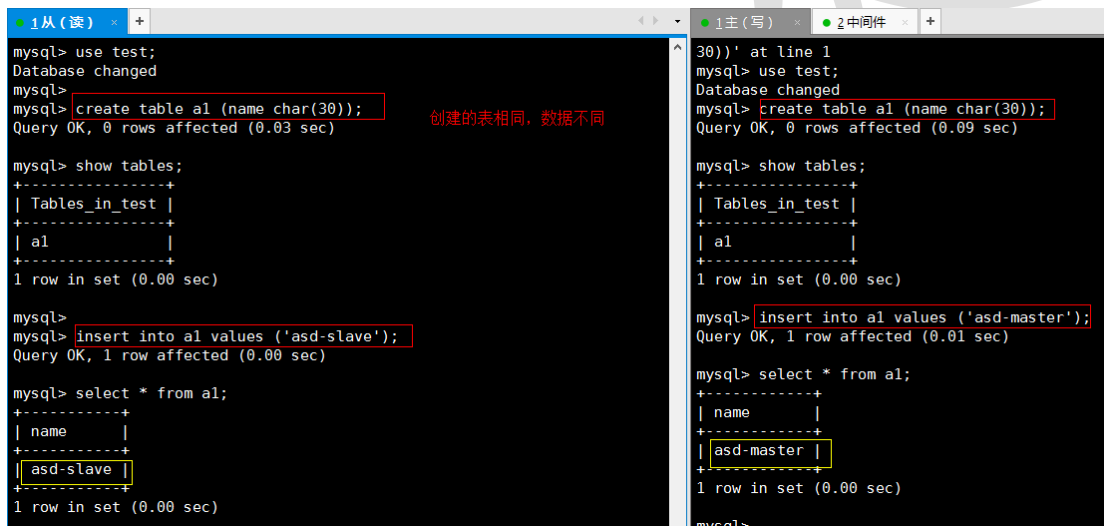
Copyright (c) 2000, 2012, Oracle and/or its affiliates. All rights reserved.

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>
```

在主、从服务器上创建表 a1，在主服务器的表中插入数据



```
mysql> use test;
Database changed
mysql> create table a1 (name char(30));
Query OK, 0 rows affected (0.03 sec)

mysql> show tables;
+-----+
| Tables_in_test |
+-----+
| a1              |
+-----+
1 row in set (0.00 sec)

mysql> insert into a1 values ('asd-slave');
Query OK, 1 row affected (0.00 sec)

mysql> select * from a1;
+-----+
| name      |
+-----+
| asd-slave |
+-----+
1 row in set (0.00 sec)

mysql> use test;
Database changed
mysql> create table a1 (name char(30));
Query OK, 0 rows affected (0.09 sec)

mysql> show tables;
+-----+
| Tables_in_test |
+-----+
| a1              |
+-----+
1 row in set (0.00 sec)

mysql> insert into a1 values ('asd-master');
Query OK, 1 row affected (0.01 sec)

mysql> select * from a1;
+-----+
| name      |
+-----+
| asd-master |
+-----+
1 row in set (0.00 sec)

mysql>
```

之后在客户端登录测试
读取池的效果：

```
1 client × +
mysql>
mysql> use test;
Database changed
mysql> show tables;
+-----+
| Tables_in_test |
+-----+
| a1              |
+-----+
1 row in set (0.01 sec)

mysql>
mysql> select * from a1;
+-----+
| name      |
+-----+
| asd-master |
+-----+
1 row in set (0.02 sec)

mysql> select * from a1;
+-----+
| name      |
+-----+
| asd-slave  |
+-----+
1 row in set (0.03 sec)

mysql> 
```

轮次出现，读是1 2 两台

写入池效果

```
1 client x +
mysql>
mysql> insert into a1 values ('asd-amoeba-01');
Query OK, 1 row affected (0.04 sec)

mysql> insert into a1 values ('asd-amoeba-02');
Query OK, 1 row affected (0.03 sec)

mysql> insert into a1 values ('asd-amoeba-03');
Query OK, 1 row affected (0.02 sec)

mysql> select * from a1;
+-----+
| name          |
+-----+
| asd-master    |
| asd-amoeba-01 |
| asd-amoeba-02 |
| asd-amoeba-03 |
+-----+
4 rows in set (0.01 sec)

mysql>
mysql> select * from a1;
+-----+
| name          |
+-----+
| asd-slave     |
+-----+
1 row in set (0.05 sec)
```

模拟用户逛淘宝下单，是写入的操作。。

写入的数据和主（写）服务器上的数据一起，而在从（读）服务器上没有，说明读写分离成功

以上测试纯粹为了实验效果，在实际生产中，主从开启，主服务器上写入的数据也会同步到从服务器中