

MySQL 主从同步与读写分离配置图文详解

目录

一、 实验目的.....	2
二、 基本环境.....	2
三、 主从同步.....	3
1、 安装与初始化数据库.....	3
2、 授权远程登录.....	4
3、 创建测试数据库.....	4
4、 主从同步配置.....	5
5、 打包同步数据库并送至 slave.....	6
6、 修改 slave 数据库配置文件.....	6
7、 测试主从同步.....	7
四、 读写分离.....	8
1、 下载并解压 amoeba	8
2、 修改配置文件.....	8
3、 运行软件.....	10
4、 测试读写分离.....	11
五、 综合测试.....	13
六、 故障排查.....	14
1、 JVM 启动失败.....	14
2、 主从未同步.....	15

欢迎点击这里的链接进入精彩的[Linux公社](http://www.Linuxidc.com)网站

Linux公社（www.Linuxidc.com）于2006年9月25日注册并开通网站，Linux现在已经成为一种广受关注和支持的一种操作系统，IDC是互联网数据中心，LinuxIDC就是关于Linux的数据中心。

[Linux公社](http://www.Linuxidc.com)是专业的Linux系统门户网站，实时发布最新Linux资讯，包括Linux、Ubuntu、Fedora、RedHat、红旗Linux、Linux教程、Linux认证、SUSE Linux、Android、Oracle、Hadoop、CentOS、MySQL、Apache、Nginx、Tomcat、Python、Java、C语言、OpenStack、集群等技术。

Linux公社（LinuxIDC.com）设置了有一定影响力的Linux专题栏目。

Linux公社 主站网址：www.linuxidc.com 旗下网站：www.linuxidc.net

包括：[Ubuntu 专题](#) [Fedora 专题](#) [Android 专题](#) [Oracle 专题](#) [Hadoop 专题](#)
[RedHat 专题](#) [SUSE 专题](#) [红旗 Linux 专题](#) [CentOS 专题](#)



Linux 公社微信公众号：[linuxidc_com](#)



微信扫一扫

Linuxidc.com

订阅专业的最新Linux资讯及开源技术教程。

搜索微信公众号：[linuxidc_com](#)

MySQL 主从同步与读写分离

一、实验目的

实际的生产环境中，由单台 MySQL 作为独立的数据库是完全不能满足实际需求的，无论是在安全性，高可用性以及高并发等各个方面。通过主从同步（Master-Slave）的方式来同步数据，再通过读写分离（amobe）来提升数据库的并发负载能力。

二、基本环境

系统：Red Hat Enterprise Linux 6.3 32 位、MySQL 5.1、amobe 3.0.5、java1.6

两天主机：

Master IP: 192.168.1.10 主机名: xuel_master_write

Slave IP: 192.168.1.11 主机名: xuel_slave_read

初始配置：关闭防火墙或将 3306 与 8066 添加例外，关闭 Selinux。

```
[root@xuel-slave-read ~]# cat /etc/issue
Red Hat Enterprise Linux Server release 6.3 (Santiago)
Kernel \r on an \m

[root@xuel-slave-read ~]# uname -a
Linux xuel-slave-read 2.6.32-279.el6.i686 #1 SMP Wed Jun 13 18:23:32 EDT 2012 i686 i686 i386 GNU/Linux
```

```
ssh://192.168.1.11:22
要添加当前会话，点击左侧的箭头按钮。

1 master-sql-write x 2 slave-sql-read x +

[root@xuel-slave-read ~]# ifconfig
eth1      Link encap:Ethernet  HWaddr 00:0C:29:C8:4A:36
          inet addr:192.168.1.11  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fec8:4a36/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:194 errors:0 dropped:0 overruns:0 frame:0
          TX packets:109 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:18019 (17.5 KiB)  TX bytes:13672 (13.3 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:32 errors:0 dropped:0 overruns:0 frame:0
          TX packets:32 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:2276 (2.2 KiB)  TX bytes:2276 (2.2 KiB)
```

```

ssh://192.168.1.11:22
要添加当前会话，点击左侧的箭头按钮。
1 master-sql-write x 2 slave-sql-read x +
[root@xuel-slave-read ~]# ifconfig
eth1      Link encap:Ethernet  HWaddr 00:0C:29:C8:4A:36
          inet addr:192.168.1.11  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fec8:4a36/64  Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:194 errors:0 dropped:0 overruns:0 frame:0
          TX packets:109 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:18019 (17.5 KiB)  TX bytes:13672 (13.3 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128  Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:32 errors:0 dropped:0 overruns:0 frame:0
          TX packets:32 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:2276 (2.2 KiB)  TX bytes:2276 (2.2 KiB)

```

三、主从同步

1、安装与初始化数据库

安装 mysql，并设置开机自启动

`yum install -y mysql mysql-devel mysql-server`

`/etc/init.d/mysqld start`

`chkconfig mysqld on`

设置 master 数据库密码为: mysqlmaster

设置 slave 数据库密码为: mysqlslave

```

[root@xuel-master-write ~]# mysqladmin -u root password "mysqlmaster"
[root@xuel-master-write ~]# mysql -uroot -pmysqlmaster
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3
Server version: 5.1.61 Source distribution

Copyright (c) 2000, 2011, 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>

```

```
[root@xuel-slave-read ~]# mysqladmin -u root password "mysqlslave"
[root@xuel-slave-read ~]# mysql -uroot -pmysqlslave
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3
Server version: 5.1.61 Source distribution

Copyright (c) 2000, 2011, 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>
```

2、授权远程登录

分别赋予 mysql 远程登录权限

grant all privileges on 数据库.表 to 用户@"IP" identified by "密码";

```
[root@xuel-master-write ~]# mysqladmin -u root password "mysqlmaster"
[root@xuel-master-write ~]# mysql -uroot -pmysqlmaster
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3
Server version: 5.1.61 Source distribution

Copyright (c) 2000, 2011, 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> grant all privileges on *.* to root@"192.168.1.11" identified by "mysqlmaster";
Query OK, 0 rows affected (0.00 sec)

mysql> flush privileges;
Query OK, 0 rows affected (0.00 sec)
```

```
[root@xuel-slave-read ~]# mysql -uroot -pmysqlslave
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 5.1.61 Source distribution

Copyright (c) 2000, 2011, 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> grant all privileges on *.* to root@"192.168.1.10" identified by "mysqlslave";
Query OK, 0 rows affected (0.00 sec)

mysql> flush privileges;
Query OK, 0 rows affected (0.00 sec)
```

3、创建测试数据库

在 master 创建数据库 master_test,并插入数据

create database 数据库名称; //创建数据库

Use 数据库; //改变所使用的数据库

create table 表名称 (字段名称 数据类型); //创建表

description 表名称;

//查看表结构

Select 字段 from 表;

//数据库查询

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

mysql> use master_test;
Database changed
mysql> create table test1(id int,name varchar(24));
Query OK, 0 rows affected (0.01 sec)

mysql> insert into test1 values(1,"xuel");
Query OK, 1 row affected (0.00 sec)

mysql> desc test1;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | int(11)       | YES  |     | NULL    |       |
| name  | varchar(24)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> select * from test1;
+-----+-----+
| id  | name |
+-----+-----+
| 1   | xuel |
+-----+-----+
1 row in set (0.00 sec)
```

查看 master 目前数据库如下:

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

4、主从同步配置

本次实验同步 master_test 数据库

修改 mysql 配置文件 my.cnf 如下

```
[root@xuel-master-write ~]# cat /etc/my.cnf
[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
server-id=1
log-bin=mysql-bin
binlog-do-db=master_test
binlog-ignore-db=mysql

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

server-id = 1 //数据库 ID 号, 为 1 时表示为
 log-bin=mysql-bin //启用二进制日志;
 binlog-do-db=master_test//需要同步的二进制数据库名, 可以写多个;
 binlog-ignore-db=mysql //不同步的二进制数据库名
 重启服务器查看 master 状态

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

```
mysql> show master status \G;
***** 1 row *****
      File: mysql-bin.000002
      Position: 106
    Binlog_Do_DB: master_test
    Binlog_Ignore_DB: mysql
1 row in set (0.00 sec)
```

5、打包同步数据库并送至 slave

打包 master 服务器 master_test 数据库

```
[root@xuel-master-write ~]# cd /var/lib/mysql/
[root@xuel-master-write mysql]# ls
ibdata1 ib_logfile0 ib_logfile1 master_test mysql mysql-bin.000001 mysql-bin.index mysql.sock test
[root@xuel-master-write mysql]# tar -zcvf master_test.tar.gz master_test/
master_test/
master_test/test1.MYD
master_test/test1.frm
master_test/test1.MYI
master_test/db.opt
```

将其拷贝到 slave 服务器

```
[root@xuel-master-write mysql]# scp master_test.tar.gz root@192.168.1.11:/var/lib/mysql/
The authenticity of host '192.168.1.11 (192.168.1.11)' can't be established.
RSA key fingerprint is 1e:ad:c2:cc:ed:5d:ad:ce:9b:06:00:cd:da:8b:9b:58.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.11' (RSA) to the list of known hosts.
root@192.168.1.11's password:
master_test.tar.gz 100% 576 0.6KB/s 00:00
```

6、修改 slave 数据库配置文件

修改 slave 服务器 mysql 配置文件

```
[root@xuel-slave-read ~]# cat /etc/my.cnf
[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
server-id=2
log-bin=mysql-bin
master-host=192.168.1.10
master-user=root
master-password=mysqlmaster
master-port=3306

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

解压拷贝过来的 master_test 数据库

```

[root@xuel-slave-read mysql]# tar -xvf master_test.tar.gz
master_test/
master_test/test1.MYD
master_test/test1.frm
master_test/test1.MYI
master_test/db.opt
[root@xuel-slave-read mysql]# ls
ibdata1      master.info      mysql           mysqld-relay-bin.000001  relay-log.info
ib_logfile0  master_test      mysql-bin.000001  mysqld-relay-bin.index  test
ib_logfile1  master_test.tar.gz  mysql-bin.index  mysql.sock

```

重启 mysql 服务并查看 slave 状态。

```

mysql> show slave status \G;
***** 1. row *****
Slave_IO_State: Waiting for master to send event
Master_Host: 192.168.1.10
Master_User: root
Master_Port: 3306
Connect_Retry: 60
Master_Log_File: mysql-bin.000002
Read_Master_Log_Pos: 106
Relay_Log_File: mysqld-relay-bin.000004
Relay_Log_Pos: 251
Relay_Master_Log_File: mysql-bin.000002
Slave_IO_Running: Yes
Slave_SQL_Running: Yes
Replicate_Do_DB:
Replicate_Ignore_DB:
Replicate_Do_Table:
Replicate_Ignore_Table:
Replicate_Wild_Do_Table:
Replicate_Wild_Ignore_Table:
Last_Errno: 0
Last_Error:
Skip_Counter: 0
Exec_Master_Log_Pos: 106
Relay_Log_Space: 407
Until_Condition: None

```

7、测试主从同步

Master 服务器 master_test 新建表测试。

```

mysql> use master 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_master_test |
+-----+
| test1                  |
+-----+
1 row in set (0.00 sec)

mysql> create table test2(id int,sex varchar(4));
Query OK, 0 rows affected (0.05 sec)


mysql> show tables;
+-----+
| Tables_in_master_test |
+-----+
| test1                  |
| test2                  |
+-----+

```


查看 slave 服务器

```
mysql> use master_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_master_test |
+-----+
| test1                  |
| test2                  |
+-----+
2 rows in set (0.00 sec)
```



```
mysql> desc test2;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | int(11)       | YES  |     | NULL    |       |
| sex   | varchar(4)    | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

测试已经完成主从同步。

四、读写分离

本次实验我们使用 Amoeba(变形虫) 这个开源项目来完成 MySQL 的读写分离。这个软件致力于 MySQL 的分布式数据库前端代理层,它主要在应用层访问 MySQL 的时候充当 SQL 路由功能,专注于分布式数据库代理层(Database Proxy)开发。座落与 Client、DB Server(s) 之间,对客户端透明。具有负载均衡、高可用性、SQL 过滤、读写分离、可路由相关的到目标数据库、可并发请求多台数据库合并结果。其基于 java 语言开发,所以需要安装 jdk 运行环境来支持软件运行。

1、下载并解压 amoeba

将下载好的 amoeba 解压

```
[root@xuel-master-write src]# ls
amoeba-mysql-3.0.5-RC-distribution.zip  debug  kernels  linux
[root@xuel-master-write src]# unzip amoeba-mysql-3.0.5-RC-distribution.zip
Archive:  amoeba-mysql-3.0.5-RC-distribution.zip
  creating: amoeba-mysql-3.0.5-RC/
  creating: amoeba-mysql-3.0.5-RC/lib/
  inflating: amoeba-mysql-3.0.5-RC/lib/amoeba-core-3.0.5-RC.jar
  inflating: amoeba-mysql-3.0.5-RC/lib/log4j-1.2.12.jar
  inflating: amoeba-mysql-3.0.5-RC/lib/commons-lang-2.4.jar
  inflating: amoeba-mysql-3.0.5-RC/lib/commons-logging-1.1.1.jar
  inflating: amoeba-mysql-3.0.5-RC/lib/commons-beanutils-1.8.0.jar
  inflating: amoeba-mysql-3.0.5-RC/lib/commons-collections-3.2.1.jar
  inflating: amoeba-mysql-3.0.5-RC/lib/ognl-3.0.1.jar
  inflating: amoeba-mysql-3.0.5-RC/lib/javassist-3.11.0.GA.jar
  inflating: amoeba-mysql-3.0.5-RC/lib/toolkit-common-pool-1.2.0.jar
  inflating: amoeba-mysql-3.0.5-RC/lib/commons-pool-1.6.jar
```

2、修改配置文件

主要修改两个文件实现读写分离

```
[root@xuel-master-write src]# ls
amoeba-mysql-3.0.5-RC  amoeba-mysql-3.0.5-RC-distribution.zip  debug  kernels  linux
[root@xuel-master-write src]# cd amoeba-mysql-3.0.5-RC
[root@xuel-master-write amoeba-mysql-3.0.5-RC]# ls
benchmark  bin  conf  jvm.properties  lib
[root@xuel-master-write amoeba-mysql-3.0.5-RC]# cd conf/
[root@xuel-master-write conf]# ls
access_list.conf  amoeba.xml  dbServers.xml  functionMap.xml  log4j.xml  ruleFunctionMap.xml
amoeba.dtd        dbserver.dtd  function.dtd  log4j.dtd        rule.dtd    rule.xml
```

查看已经默认安装 java-1.6.0, 如果未安装需要安装 java

```
[root@xuel-master-write conf]# rpm -qa | grep java
java-1.6.0-openjdk-1.6.0.0-1.45.1.11.1.el6.i686
tzdata-java-2012c-1.el6.noarch
```

修改 amoeba.xml 文件

Amoeba 软件运行监听端口: 8066, 在 master 之上配置

```
<property name="port">8066</property>

<property name="ipAddress">192.168.1.10</property>

<property name="connectionFactory">
    <bean class="com.meidusa.amoeba.mysql.net.MysqlClientConnectionFactory">
        <property name="sendBufferSize">128</property>
        <property name="receiveBufferSize">64</property>
    </bean>
</property>
```

设置连接 amoeba 的用户名和密码:

```
<property name="authenticateProvider">
    <bean class="com.meidusa.amoeba.mysql.server.MysqlClientAuthenticator">

        <property name="user">xuel</property>

        <property name="password">8gecco4</property>

        <property name="filter">
            <bean class="com.meidusa.toolkit.net.authenticate.server.IPAccessController">
                <property name="ipFile">${amoeba.home}/conf/access_list.conf</property>
            </bean>
        </property>
    </bean>
</property>
```

配置默认 pool 与读写池

```
<queryRouter class="com.meidusa.amoeba.mysql.parser.MysqlQueryRouter">
    <property name="ruleLoader">
        <bean class="com.meidusa.amoeba.route.TableRuleFileLoader">
            <property name="ruleFile">${amoeba.home}/conf/rule.xml</property>
            <property name="functionFile">${amoeba.home}/conf/ruleFunctionMap.xml</property>
        </bean>
    </property>
    <property name="sqlFunctionFile">${amoeba.home}/conf/functionMap.xml</property>
    <property name="LRUMapSize">1500</property>
    <property name="defaultPool">readpool</property>
    <property name="writePool">writepool</property>
    <property name="readPool">readpool</property>
    <property name="needParse">true</property>
</queryRouter>
```

修改 dbServers.xml

Master 测试数据库

```

<factoryConfig class="com.meidusa.amoeba.mysql.net.MysqlServerConnectionFactory">
  <property name="connectionManager">${defaultManager}</property>
  <property name="sendBufferSize">64</property>
  <property name="receiveBufferSize">128</property>

  <!-- mysql port -->
  <property name="port">3306</property>

  <!-- mysql schema -->
  <property name="schema">test</property>

  <!-- mysql user -->
  <property name="user">root</property>

  <property name="password">mysqlmaster</property>
</factoryConfig>

```

定义 server1 为 master、server2 为 slave

```

<dbServer name="server1" parent="abstractServer">
  <factoryConfig>
    <!-- mysql ip -->
    <property name="ipAddress">192.168.1.10</property>
    <property name="user">root</property>
    <property name="password">mysqlmaster</property>
  </factoryConfig>
</dbServer>

<dbServer name="server2" parent="abstractServer">
  <factoryConfig>
    <!-- mysql ip -->
    <property name="ipAddress">192.168.1.11</property>
    <property name="user">root</property>
    <property name="password">mysqlslave</property>
  </factoryConfig>
</dbServer>

```

定义 writepool 为 server1, readpool 为 server1、server2、server1

<!-- Load balancing strategy: 1=ROUNDROBIN , 2=WEIGHTBASED , 3=HA-->

所在 pool 的数据库轮训规则, 支持轮训、权重、HA

```

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

    <!-- Separated by commas,such as: server1,server2,server1 -->
    <property name="poolNames">server1</property>
  </poolConfig>
</dbServer>

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

    <!-- Separated by commas,such as: server1,server2,server1 -->
    <property name="poolNames">server1,server2,server1</property>
  </poolConfig>
</dbServer>

```

配置 java 环境修改 bin/launcher 文件加入 JAVA_HOME=/usr

```

JAVA_HOME=/usr
cygwin=false;
case "`uname`" in
  CYGWIN*) cygwin=true ;;
esac

```

3、运行软件

```

[root@xuel-master-write bin]# ./launcher
2016-07-09 09:22:01 [INFO] Project Name=Amoeba-MySQL, PID=2816 , starting...
log4j:WARN log4j config load completed from file:/usr/src/amoeba-mysql-3.0.5-RC/conf/log4j.xml
2016-07-09 09:22:03,196 INFO context.MySqlRuntimeContext - Amoeba for MySQL current version=5.1.45-mysql-amoeba-proxy-3.0.4-BETA
log4j:WARN ip access config load completed from file:/usr/src/amoeba-mysql-3.0.5-RC/conf/access_list.conf
2016-07-09 09:22:03,638 INFO net.ServerableConnectionManager - Server listening on /192.168.1.10:8066.

```

查看可以正常监听端口

```

[root@xuel-master-write ~]# netstat -lntp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:48197          0.0.0.0:*               LISTEN      1273/rpc.statd
tcp        0      0 0.0.0.0:3306           0.0.0.0:*               LISTEN      2597/mysqld
tcp        0      0 0.0.0.0:111            0.0.0.0:*               LISTEN      1255/rpcbind
tcp        0      0 0.0.0.0:22             0.0.0.0:*               LISTEN      1466/sshd
tcp        0      0 127.0.0.1:631          0.0.0.0:*               LISTEN      1351/cupsd
tcp        0      0 127.0.0.1:25           0.0.0.0:*               LISTEN      1542/master
tcp        0      0 0.0.0.0:8066           0.0.0.0:*               LISTEN      2816/java
tcp        0      0 0.0.0.0:45803          0.0.0.0:*               LISTEN      1273/rpc.statd
tcp        0      0 0.0.0.0:111            0.0.0.0:*               LISTEN      1255/rpcbind
tcp        0      0 0.0.0.0:22             0.0.0.0:*               LISTEN      1466/sshd
tcp        0      0 0.0.0.0:1:631          0.0.0.0:*               LISTEN      1351/cupsd
tcp        0      0 0.0.0.0:1:25           0.0.0.0:*               LISTEN      1542/master

```

4、测试读写分离

在 master 新建数据库

```

[root@xuel-master-write ~]# mysql -uroot -pmysqlmaster;
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 409
Server version: 5.1.61-log Source distribution

Copyright (c) 2000, 2011, 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> create database master1;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| master1 |
| master_test |
| mysql |
| test |
+-----+
5 rows in set (0.01 sec)

```

Slave 新建数据库 slave1

```
[root@xuel-slave-read mysql]# mysql -uroot -pmysqlslave
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 7
Server version: 5.1.61-log Source distribution

Copyright (c) 2000, 2011, 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> create database slave1;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| master_test |
| mysql |
| slave1 |
| test |
+-----+
```

通过 slave 连接测试

测试读，分别从 master 和 slave 服务器上面读数据库。

```
[root@xuel-slave-read ~]# mysql -uxuel -p8gecco4 -h 192.168.1.10 -P 8066
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 18782306
Server version: 5.1.45-mysql-amoeba-proxy-3.0.4-BETA Source distribution

Copyright (c) 2000, 2011, 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 |
| master_test |
| mysql |
| slave1 |
| test |
+-----+
5 rows in set (0.01 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| master1 |
| master_test |
| mysql |
| test |
+-----+
5 rows in set (0.00 sec)
```

测试写如数据：

```
mysql> create database xuel_test_write;  
Query OK, 1 row affected (0.04 sec)
```

```
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| master1 |  
| master_test |  
| mysql |  
| test |  
| xuel_test_write |  
+-----+  
6 rows in set (0.00 sec)
```

Master

```
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| master1 |  
| master_test |  
| mysql |  
| test |  
| xuel_test_write |  
+-----+  
6 rows in set (0.00 sec)
```

Slave

```
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| master_test |  
| mysql |  
| slave1 |  
| test |  
+-----+  
5 rows in set (0.00 sec)
```

测试读写分离成功。

五、综合测试

```
mysql> use master_test;
Database changed
mysql> show tables;
+-----+
| Tables_in_master_test |
+-----+
| test1                  |
| test2                  |
+-----+
2 rows in set (0.00 sec)

mysql> create table test_read_write(id int);
Query OK, 0 rows affected (0.05 sec)

mysql> show tables;
+-----+
| Tables_in_master_test |
+-----+
| test1                  |
| test2                  |
| test_read_write        |
+-----+
3 rows in set (0.00 sec)
```

Master 查看:

```
mysql> use master_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_master_test |
+-----+
| test1                  |
| test2                  |
| test_read_write        |
+-----+
3 rows in set (0.00 sec)
```

Slave 查看:

```
mysql> show tables;
+-----+
| Tables_in_master_test |
+-----+
| test1                  |
| test2                  |
| test_read_write        |
+-----+
3 rows in set (0.00 sec)
```

至此数据库读写分离已经配置完成。

六、故障排查

1、JVM 启动失败

stack size 太小，导致 JVM 启动失败

```
[root@xuel-master-write bin]# ./launcher
OpenJDK 64-Bit Server VM warning: ignoring option PermSize=16m; support was removed in 8.0
OpenJDK 64-Bit Server VM warning: ignoring option MaxPermSize=96m; support was removed in 8.0

The stack size specified is too small, Specify at least 228k
Error: Could not create the Java Virtual Machine.
Error: A fatal exception has occurred. Program will exit.
OpenJDK 64-Bit Server VM warning: ignoring option PermSize=16m; support was removed in 8.0
OpenJDK 64-Bit Server VM warning: ignoring option MaxPermSize=96m; support was removed in 8.0

The stack size specified is too small, Specify at least 228k
Error: Could not create the Java Virtual Machine.
Error: A fatal exception has occurred. Program will exit.
```

解决方法：修改 amoeba-mysql-3.0.5-RC/jvm.properties

将原有

```
JVM_OPTIONS="-server -Xms256m -Xmx1024m -Xss196k -XX:PermSize=16m -XX:MaxPermSize=96m"
```

修改为：

```
JVM_OPTIONS="-server -Xms1024m -Xmx1024m -Xss256k -XX:PermSize=16m -XX:MaxPermSize=96m"
```

2、主从未同步

解决方法：查看 master 状态

```
mysql> show master status \G;
***** 1. row *****
      File: mysql-bin.000002
      Position: 106
    Binlog_Do_DB: master_test
    Binlog_Ignore_DB: mysql
```

在 slave 之上运行

```
mysql> slave stop;
```

```
mysql> change master to Master_Log_File='mysql-bin.000002',Master_Log_Pos=106;
```

```
mysql> slave start;
```

```
mysql> show slave status\G
```