一、环境

1.linux版本：CentOS release 6.8 (Final)

2.mysql版本：mysql-5.7.30-1.el6.x86_64.rpm-bundle.tar

3.架构

4台主机，1主、2从、1 MHA

master:192.168.238.156

slave: 192.168.238.157, 192.168.238.158

MHA: 192.168.238.159

二、mysql安装

1.移除原系统mysql和mariadb

1)查看系统是否安装mariadb和mysql

rpm -qa|grep -i mariadb

rpm -qa|grep -i mysql

2)移除

```
rpm -e MySQL-devel-5.6.34-1.el6.i686 --nodeps
rpm -e MySQL-server-5.6.34-1.el6.i686 --nodeps
rpm -e MySQL-client-5.6.34-1.el6.i686 --nodeps
```

2.rpm方式安装mysql

1)解压缩mysql tar包

```
tar -xvf mysql-5.7.30-1.el6.x86_64.rpm-bundle.tar
```

2)依次安装common/libs/libs-compat/client/server/devel rpm

```
rpm -ivh mysql-community-common-5.7.30-1.el6.x86_64.rpm
rpm -ivh mysql-community-libs-5.7.30-1.el6.x86_64.rpm
rpm -ivh mysql-community-libs-compat-5.7.30-1.el6.x86_64.rpm
rpm -ivh mysql-community-client-5.7.30-1.el6.x86_64.rpm
rpm -ivh mysql-community-server-5.7.30-1.el6.x86_64.rpm
rpm -ivh mysql-community-devel-5.7.30-1.el6.x86_64.rpm
```

3)初始化mysql

mysqld --initialize --user=mysql

4)查看mysql初始化密码

cat /var/log/mysqld.log

5)启动mysql

service mysql start

6)登录重置密码

set password=password('root');

7)重新登录

三、mysql主从环境搭建

1.主库配置（master:192.168.238.156)

1)修改 /etc/my.cnf

vim /etc/my.cnf

```
[mysqld]
log_bin=mysql-bin
server-id=1
relay_log=mysql-relay-bin
sync-binlog=1

#对哪些库不同步
binlog-ignore-db=information_schema
binlog-ignore-db=performance_schema
binlog-ignore-db=sys

#对哪些库同步
#binlog-do-db=lagou
```

2)重启mysql

service mysqld restart

3)登录mysql授权

```
grant replication slave on *.* to 'root'@'%' identified by 'root';
grant all privileges on *.* to 'root'@'%' identified by 'root';
flush privileges;
```

4)查看主库状态

show master status

```
mysql> show master status;
+------------------+----------+--------------+-------------------------------------------+------------------+
| File             | Position | Binlog_Do_DB | Binlog_Ignore_DB                          | Executed_Gtid_Set |
+------------------+----------+--------------+-------------------------------------------+------------------+
| mysql-bin.000001 |      869 |              | information_schema,performance_schema,sys |                  |
+------------------+----------+--------------+-------------------------------------------+------------------+
1 row in set (0.00 sec)
```

2.从库配置（slave: 192.168.238.157)

1)修改 /etc/my.cnf

vim /etc/my.cnf

```
[mysqld]
server-id=2
log_bin=mysql-bin
relay_log=mysql-relay-bin
read_only=1
relay_log_purge=0
```

```
#对哪些库不同步
binlog-ignore-db=information_schema
binlog-ignore-db=performance_schema
binlog-ignore-db=sys

#对哪些库同步
#binlog-do-db=lagou
```

2)重启mysql

　　service mysqld restart

3)登录mysql查看从库状态
　　show slave status \G

4)同步主库
　　change master to
master_host='192.168.238.156',master_port=3306,master_user='root',master_password='root',m
aster_log_file='mysql-bin.000001',master_log_pos=869;

5)开启从库
　　start slave;
6)查看从库状态
　　show slave status \G


```
mysql> show slave status \G
Empty set (0.00 sec)

mysql> change master to master_host='192.168.238.156',master_port=3306,master_user='root',master_password='root',master_log_file='mysql-bin.000001',master_log_pos=869;
Query OK, 0 rows affected, 2 warnings (0.09 sec)

mysql> start slave;
Query OK, 0 rows affected (0.05 sec)

mysql> show slave status \G
*************************** 1. row ***************************
               Slave_IO_State: Checking master version
                  Master_Host: 192.168.238.156
                  Master_User: root
                  Master_Port: 3306
                Connect_Retry: 60
              Master_Log_File: mysql-bin.000001
          Read_Master_Log_Pos: 869
               Relay_Log_File: mysql-relay-bin.000001
                Relay_Log_Pos: 4
        Relay_Master_Log_File: mysql-bin.000001
             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
```

3.从库配置（slave: 192.168.238.158）

同从库(192.168.238.157)配置， /etc/my.cnf 中server-id=3

4.测试主从配置

主库执行sql，查看从库是否同步

```
create database lagou;
show database;
use lagou;
create table position (id int primary key auto_increment, name varchar(20),
salary varchar(20), city varchar(20)) engine=innodb charset=utf8;
create table position_detail (id int primary key auto_increment, pid int,
description text) engine=innodb charset=utf8;
```

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| lagou              |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)

mysql> use lagou;
Database changed
mysql> create table position (id int primary key auto_increment, name varchar(20), salary varchar(20), city varchar(20)) engine=innodb charset=utf8;
Query OK, 0 rows affected (0.09 sec)

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

mysql> create table position_detail (id int primary key auto_increment, pid int, description text) engine=innodb charset=utf8;
Query OK, 0 rows affected (0.03 sec)

mysql> show tables;
+----------------+
| Tables_in_lagou |
+----------------+
| position        |
| position_detail |
+----------------+
2 rows in set (0.01 sec)
```

```
              Auto_Position: 0
         Replicate_Rewrite_DB:
               Channel_Name:
          Master_TLS_Version:
1 row in set (0.00 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| lagou              |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)

mysql> use lagou;
Database changed
mysql> show tables;
+----------------+
| Tables_in_lagou |
+----------------+
| position       |
+----------------+
1 row in set (0.00 sec)

mysql> show tables;
+----------------+
| Tables_in_lagou |
+----------------+
| position        |
| position_detail |
+----------------+
2 rows in set (0.00 sec)
```

● 1 centos_192.168.238.156 ×   ● 2 centos_192.168.238.157 ×   ● 3 centos_192.168.238.... ×   +

```
                Auto_Position: 0
        Replicate_Rewrite_DB:
                Channel_Name:
           Master_TLS_Version:
1 row in set (0.00 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| lagou              |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)

mysql> use lagou;
Database changed
mysql> show tables;
+-----------------+
| Tables_in_lagou |
+-----------------+
| position        |
+-----------------+
1 row in set (0.00 sec)

mysql> show tables;
+-----------------+
| Tables_in_lagou |
+-----------------+
| position        |
| position_detail |
+-----------------+
2 rows in set (0.00 sec)
```

5.半同步复制机制配置

主库配置

1)登录mysql后查看是否有动态加载
  select @@have_dynamic_loading;

2)查看插件
  show plugins;

3)安装插件
  install plugin rpl_semi_sync_master soname 'semisync_master.so';

4)查看
  show variables like '%semi%';

5)设置rpl_semi_sync_master_enabled和rpl_semi_sync_master_timeout
  set global rpl_semi_sync_master_enabled=1;
  set global rpl_semi_sync_master_timeout=1000;

```
mysql> install plugin rpl_semi_sync_master soname 'semisync_master.so';
Query OK, 0 rows affected (0.05 sec)

mysql> show variables like '%semi%';
+-------------------------------------------+------------+
| Variable_name                             | Value      |
+-------------------------------------------+------------+
| rpl_semi_sync_master_enabled              | OFF        |
| rpl_semi_sync_master_timeout              | 10000      |
| rpl_semi_sync_master_trace_level          | 32         |
| rpl_semi_sync_master_wait_for_slave_count | 1          |
| rpl_semi_sync_master_wait_no_slave        | ON         |
| rpl_semi_sync_master_wait_point           | AFTER_SYNC |
+-------------------------------------------+------------+
6 rows in set (0.05 sec)

mysql> set global rpl_semi_sync_master_enabled=1;
Query OK, 0 rows affected (0.00 sec)

mysql> set global rpl_semi_sync_master_timeout=1000;
Query OK, 0 rows affected (0.00 sec)

mysql> show variables like '%semi%';
+-------------------------------------------+------------+
| Variable_name                             | Value      |
+-------------------------------------------+------------+
| rpl_semi_sync_master_enabled              | ON         |
| rpl_semi_sync_master_timeout              | 1000       |
| rpl_semi_sync_master_trace_level          | 32         |
| rpl_semi_sync_master_wait_for_slave_count | 1          |
| rpl_semi_sync_master_wait_no_slave        | ON         |
| rpl_semi_sync_master_wait_point           | AFTER_SYNC |
+-------------------------------------------+------------+
6 rows in set (0.01 sec)
```

从库配置

1)安装插件：install plugin rpl_semi_sync_slave soname 'semisync_slave.so';

2)设置rpl_semi_sync_slave_enabled

set global rpl_semi_sync_slave_enabled=1;

3)重启slave

stop slave;

start slave;

```
mysql> install plugin rpl_semi_sync_slave soname 'semisync_slave.so';
Query OK, 0 rows affected (0.05 sec)

mysql> show variables like '%semi%';
+-----------------------------------+-------+
| Variable_name                     | Value |
+-----------------------------------+-------+
| rpl_semi_sync_slave_enabled       | OFF   |
| rpl_semi_sync_slave_trace_level   | 32    |
+-----------------------------------+-------+
2 rows in set (0.01 sec)

mysql> set global rpl_semi_sync_slave_enabled=1;
Query OK, 0 rows affected (0.00 sec)

mysql> show variables like '%semi%';
+-----------------------------------+-------+
| Variable_name                     | Value |
+-----------------------------------+-------+
| rpl_semi_sync_slave_enabled       | ON    |
| rpl_semi_sync_slave_trace_level   | 32    |
+-----------------------------------+-------+
2 rows in set (0.00 sec)
```

测试半同步复制

1主库执行sql，查看从库是否同步

```
insert into position (name, salary, city) values ('lm', '10', '北京');
insert into position_detail(pid, description) values (1, 'haha');
```

查看log日志：tail -200 /var/log/mysqld.log

```
2020-05-03T03:54:29.635475Z 5 [Warning] IP address '192.168.238.158' could not be resolved: Name or service not known
2020-05-03T03:54:29.638952Z 5 [Note] Start binlog_dump to master_thread_id(5) slave_server(3), pos(mysql-bin.000001, 869)
2020-05-03T04:09:24.972738Z 3 [Note] Semi-sync replication initialized for transactions.
2020-05-03T04:09:24.972805Z 3 [Note] Semi-sync replication enabled on the master.
2020-05-03T04:09:24.973067Z 0 [Note] Starting ack receiver thread
2020-05-03T04:18:42.410957Z 6 [Note] While initializing dump thread for slave with UUID <78ffd2a3-ad18-11ea-bd23-0050563ecf66>, found a zombie dump thread with the same UUID. Master is killing the zombie dump thread(4).
2020-05-03T04:18:42.411135Z 4 [Note] Stop asynchronous binlog_dump to slave (server_id: 2)
2020-05-03T04:18:42.411655Z 6 [Note] Start binlog_dump to master_thread_id(6) slave_server(2), pos(mysql-bin.000001, 1571)
2020-05-03T04:18:42.411782Z 6 [Note] Start semi-sync binlog_dump to slave (server_id: 2), pos(mysql-bin.000001, 1571)
2020-05-03T04:18:50.330973Z 7 [Note] While initializing dump thread for slave with UUID <54aae2ab-ad19-11ea-ad7d-005056294c78>, found a zombie dump thread with the same UUID. Master is killing the zombie dump thread(5).
2020-05-03T04:18:50.331158Z 5 [Note] Stop asynchronous binlog_dump to slave (server_id: 3)
2020-05-03T04:18:50.331685Z 7 [Note] Start binlog_dump to master_thread_id(7) slave_server(3), pos(mysql-bin.000001, 1571)
2020-05-03T04:18:50.331703Z 7 [Note] Start semi-sync binlog_dump to slave (server_id: 3), pos(mysql-bin.000001, 1571)
[root@localhost log]# tail -200 /var/log/mysqld.log
```

四、MHA安装及配置（MHA: 192.168.238.159）

1.各节点（mysql一主两从及MHA）yum源配置

```
    1)
    wget -O /etc/yum.repos.d/CentOS6-Base-163.repo
http://mirrors.163.com/.help/CentOS6-Base-163.repo
    sed -i 's/$releasever/6/g' /etc/yum.repos.d/CentOS6-Base-163.repo
    wget -O /etc/yum.repos.d/CentOS-Base-Ali.repo
http://mirrors.aliyun.com/repo/Centos-6.repo
    sed -i 's/$releasever/6/g' /etc/yum.repos.d/CentOS-Base-Ali.repo
    2)vim /etc/yum.repos.d/epel.repo
    [epel]
    name=epel
    baseurl=https://mirrors.ustc.edu.cn/epel/6Server/x86_64/
    gpgcheck=0
    enabled=1
    3)yum groupinstall 'Development tools' 'Server Platform Development' -y
```

2.配置各节点（mysql一主两从及MHA）免密码通信

```
    1)各server192.168.238.156-159
        mkdir -pv /root/.ssh
        mkdir -pv /root/rpms
    2)MHA server(192.168.238.159)
        ssh-keygen -t rsa -P ''   ##执行命令后直接回车，生成文件名默认
        cat .ssh/id_rsa.pub > .ssh/authorized_keys
        chmod 600 .ssh/authorized_keys
        scp -p .ssh/id_rsa .ssh/authorized_keys 192.168.238.156:/root/.ssh
        scp -p .ssh/id_rsa .ssh/authorized_keys 192.168.238.157:/root/.ssh
        scp -p .ssh/id_rsa .ssh/authorized_keys 192.168.238.158:/root/.ssh
        ssh 192.168.238.156 'ip addr list'
        ssh 192.168.238.157 'ip addr list'
        ssh 192.168.238.158 'ip addr list'
        ssh 192.168.238.159 'ip addr list'
```

3.安装mha4mysql-manager和mha4mysql-node

```
管理端安装(MHA:192.168.238.159)
    1)安装MHA依赖包
        yum install -y perl-DBD-MySQL
        yum install -y perl-Config-Tiny
        yum install -y perl-Log-Dispatch
        yum install -y perl-Parallel-ForkManager
        yum install -y ca-certificates
        yum install -y perl-Parallel-ForkManager
        yum install -y rrdtool perl-rrdtool rrdtool-devel perl-Params-Validate
        yum install -y perl-Time-HiRes

    2)安装mha4mysql-manager和mha4mysql-node
        rpm -ivh mha4mysql-manager-0.56-0.el6.noarch.rpm --nodeps --force
        rpm -ivh mha4mysql-node-0.56-0.el6.noarch.rpm --nodeps --force
客户端安装(192.168.238.156-158)
    yum install -y perl-DBD-MySQL
    rpm -ivh mha4mysql-node-0.56-0.el6.noarch.rpm --nodeps --force
```

4.创建MHA的工作目录和日志目录(MHA:192.168.238.159)

```
mkdir -pv /dbdata/masterha/app1
mkdir -pv /etc/masterha
```

5.vim /etc/masterha/app1.cnf(MHA:192.168.238.159)

```
[server default]
user=mhauser
password=mhapass
manager_workdir=/dbdata/masterha/app1
manager_log=/dbdata/masterha/app1/manager.log
remote_workdir=/dbdata/masterha/app1
ssh_user=root
repl_user=repluser
repl_password=replpass
ping_interval=1
[server1]
hostname=192.168.238.156
candidate_master=1
#ssh_port=22
[server2]
hostname=192.168.238.157
candidate_master=1
[server3]
hostname=192.168.238.158
#no_master=1
```

6.masterha_check

```
masterha_check_ssh --conf=/etc/masterha/app1.cnf
masterha_check_repl --conf=/etc/masterha/app1.cnf
```

```
[root@localhost mysql]# masterha_check_ssh --conf=/etc/masterha/app1.cnf
Sun Jun 14 00:38:13 2020 - [warning] Global configuration file /etc/masterha_default.cnf not found. Skipping.
Sun Jun 14 00:38:13 2020 - [info] Reading application default configuration from /etc/masterha/app1.cnf..
Sun Jun 14 00:38:13 2020 - [info] Reading server configuration from /etc/masterha/app1.cnf..
Sun Jun 14 00:38:13 2020 - [info] Starting SSH connection tests..
Sun Jun 14 00:39:40 2020 - [debug]
Sun Jun 14 00:38:14 2020 - [debug]  Connecting via SSH from root@192.168.238.157(192.168.238.157:22) to root@192.168.238.156(192.168.238.156:22)..
Warning: Permanently added '192.168.238.156' (RSA) to the list of known hosts.
Sun Jun 14 00:38:55 2020 - [debug]   ok.
Sun Jun 14 00:38:55 2020 - [debug]  Connecting via SSH from root@192.168.238.157(192.168.238.157:22) to root@192.168.238.158(192.168.238.158:22)..
Warning: Permanently added '192.168.238.158' (RSA) to the list of known hosts.
Sun Jun 14 00:39:40 2020 - [debug]   ok.
Sun Jun 14 00:39:42 2020 - [debug]
Sun Jun 14 00:38:13 2020 - [debug]  Connecting via SSH from root@192.168.238.156(192.168.238.156:22) to root@192.168.238.157(192.168.238.157:22)..
Warning: Permanently added '192.168.238.157' (RSA) to the list of known hosts.
Sun Jun 14 00:39:00 2020 - [debug]   ok.
Sun Jun 14 00:39:00 2020 - [debug]  Connecting via SSH from root@192.168.238.156(192.168.238.156:22) to root@192.168.238.158(192.168.238.158:22)..
Warning: Permanently added '192.168.238.158' (RSA) to the list of known hosts.
Sun Jun 14 00:39:42 2020 - [debug]   ok.
Sun Jun 14 00:39:46 2020 - [debug]
Sun Jun 14 00:38:14 2020 - [debug]  Connecting via SSH from root@192.168.238.158(192.168.238.158:22) to root@192.168.238.156(192.168.238.156:22)..
Warning: Permanently added '192.168.238.156' (RSA) to the list of known hosts.
Sun Jun 14 00:39:01 2020 - [debug]   ok.
Sun Jun 14 00:39:01 2020 - [debug]  Connecting via SSH from root@192.168.238.158(192.168.238.158:22) to root@192.168.238.157(192.168.238.157:22)..
Warning: Permanently added '192.168.238.157' (RSA) to the list of known hosts.
Sun Jun 14 00:39:46 2020 - [debug]   ok.
Sun Jun 14 00:39:46 2020 - [info] All SSH connection tests passed successfully.
```

```
Creating directory /dbdata/masterha/app1.. done.
  Checking slave recovery environment settings..
    Opening /var/lib/mysql/relay-log.info ... ok.
    Relay log found at /var/lib/mysql, up to mysql-relay-bin.000013
    Temporary relay log file is /var/lib/mysql/mysql-relay-bin.000013
    Testing mysql connection and privileges..mysql: [Warning] Using a password on the command line
 done.
    Testing mysqlbinlog output.. done.
    Cleaning up test file(s).. done.
Sun Jun 14 01:24:47 2020 - [info]   Executing command : apply_diff_relay_logs --command=test --sla
asterha/app1 --target_version=5.7.30-log --manager_version=0.56 --relay_log_info=/var/lib/mysql/re
Sun Jun 14 01:24:47 2020 - [info]   Connecting to root@192.168.238.158(192.168.238.158:22)..
Creating directory /dbdata/masterha/app1.. done.
  Checking slave recovery environment settings..
    Opening /var/lib/mysql/relay-log.info ... ok.
    Relay log found at /var/lib/mysql, up to mysql-relay-bin.000014
    Temporary relay log file is /var/lib/mysql/mysql-relay-bin.000014
    Testing mysql connection and privileges..mysql: [Warning] Using a password on the command line
 done.
    Testing mysqlbinlog output.. done.
    Cleaning up test file(s).. done.
Sun Jun 14 01:25:14 2020 - [info] Slaves settings check done.
Sun Jun 14 01:25:14 2020 - [info]
192.168.238.156(192.168.238.156:3306) (current master)
 +--192.168.238.157(192.168.238.157:3306)
 +--192.168.238.158(192.168.238.158:3306)

Sun Jun 14 01:25:14 2020 - [info] Checking replication health on 192.168.238.157..
Sun Jun 14 01:25:14 2020 - [info]  ok.
Sun Jun 14 01:25:14 2020 - [info] Checking replication health on 192.168.238.158..
Sun Jun 14 01:25:14 2020 - [info]  ok.
Sun Jun 14 01:25:14 2020 - [warning] master_ip_failover_script is not defined.
Sun Jun 14 01:25:14 2020 - [warning] shutdown_script is not defined.
Sun Jun 14 01:25:14 2020 - [info] Got exit code 0 (Not master dead).

MySQL Replication Health is OK.
```

7.masterha_manager --conf=/etc/masterha/app1.cnf

五、测试MHA

1.主节点(192.168.238.156)关闭mysql

   service mysql

2.从节点(192.168.238.157)查看slave状态和master状态

show slave status \G


```
mysql> show slave status \G
Empty set (0.00 sec)

mysql> show master status
    -> ;
+------------------+----------+--------------+--------------------------------------------------+-------------------+
| File             | Position | Binlog_Do_DB | Binlog_Ignore_DB                                 | Executed_Gtid_Set |
+------------------+----------+--------------+--------------------------------------------------+-------------------+
| mysql-bin.000002 |      154 |              | information_schema,performance_schema,sys        |                   |
+------------------+----------+--------------+--------------------------------------------------+-------------------+
1 row in set (0.00 sec)
```

3.从节点(192.168.238.158)查看slave状态

show slave status \G

4.原主库启动mysql, 登录后切换为从库

change master to
master_host='192.168.238.157',master_user='repluser',master_password='replpass',master_log_file='mysql-bin.000003',master_log_pos=718;

start slave;
show slave status \G

5.测试节点(192.168.238.157)插入sql

```
insert into position (name, salary, city) values ('lm', '10', '北京');
insert into position_detail(pid, description) values (11111, 'haha');

测试插入效果
select * from position;
select * from position_detail;
```

六、问题记录

1.yum提示problem making ssl connection错误

执行命令：yum install -y ca-certificates

然后重新执行报错命令

2.Host '192.168.238.159' is not allowed to connect to this MySQL server, but this is not a MySQL crash. Check MySQL server settings

mysql服务器授权

grant all on . to 'mhauser'@'192.168.238.%' identified by 'mhapass';

grant all on . to 'repluser'@'192.168.238.%' identified by 'replpass';

flush privileges;

3.[error][/usr/share/perl5/vendor_perl/MHA/ServerManager.pm, ln188] There is no alive server. We can't do failover

alter user 'mhauser'@'192.168.238.%' identified with mysql_native_password by 'mhapass';

flush privileges;

4.Binlog filtering check failed on 192.168.238.157(192.168.238.157:3306)! All log-bin enabled servers must have same binlog filtering rules (same binlog-do-db and binlog-ignore-db). Check SHOW MASTER STATUS output and set my.cnf correctly

修改从库配置 vim /etc/my.cnf

```
#对哪些库不同步
binlog-ignore-db=information_schema
binlog-ignore-db=performance_schema
binlog-ignore-db=sys

#对哪些库同步
#binlog-do-db=lagou
```

5.原主库(192.168.238.156)改为从库后，测试新主库(192.168.238.157)插入数据，原主库无数据

master_user和master_password错误：改为MHA中配置的repluser和replpass

change master to
master_host='192.168.238.157',master_user='repluser',master_password='replpass',master_log_file='mysql-bin.000003',master_log_pos=718;