**简单介绍**  
MHA（Master High Availability）目前在MySQL高可用方面是一个相对成熟的解决方案，作为MySQL高可用性环境下故障切换和主从提升的高可用软件。在MySQL故障切换过程中，MHA能做到在0~30秒之内自动完成数据库的故障切换操作，并且在进行故障切换的过程中，MHA能在最大程度上保证数据的一致性，以达到真正意义上的高可用。  
MHA由两部分组成：MHA Manager（管理节点）和MHA Node（数据节点）。MHA Manager可以单独部署在一台独立的机器上管理多个master-slave集群，也可以部署在一台slave节点上。MHA Node运行在每台MySQL服务器上，MHA Manager会定时探测集群中的master节点，当master出现故障时，它可以自动将最新数据的slave提升为新的master，然后将所有其他的slave重新指向新的master。整个故障转移过程对应用程序完全透明。  
目前MHA主要支持一主多从的架构，要搭建MHA,要求一个复制集群中必须最少有三台数据库服务器，一主二从，即一台充当master，一台充当备用master，另外一台充当从库，因为至少需要三台服务器

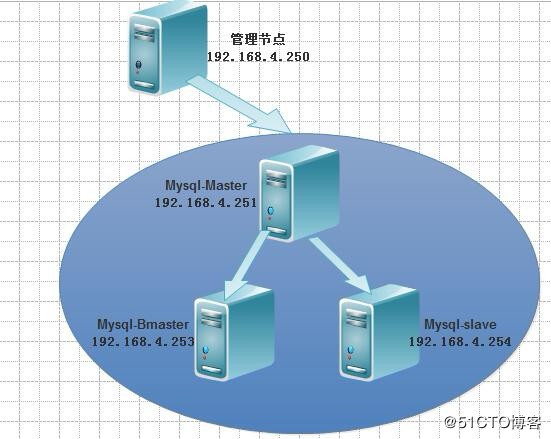
**布署规划**  
系统信息

[root@db250 ~]*# cat /etc/redhat-release*

CentOS release 6.8 (Final)

[root@db250 ~]*# uname -r*

2.6.32-642.el6.x86\_64

数据库  
mysql57-community-release-el6-9.noarch.rpm  
环境规划  
Db250 集群管理节点 192.168.4.250   
Db251 主库 192.168.4.251  
Db253 从库（备主库） 192.168.4.253  
Db254 从库 192.168.4.254  
架构图如下  


**配置SSH免密登陆与hosts文件**  
配置hosts文件

cat**>>**/etc/hosts<<EOF

192.168.4.250 db250

192.168.4.251 db251

192.168.4.253 db253

192.168.4.254 db254

EOF

配置完成后检查下

[root@db254 ~]*# tail -4 /etc/hosts*

192.168.4.250 db250

192.168.4.251 db251

192.168.4.253 db253

192.168.4.254 db254

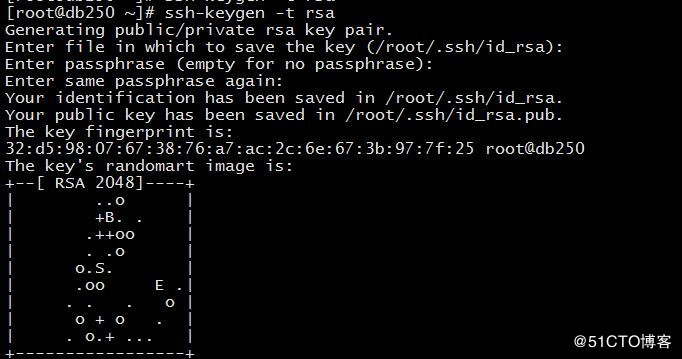
配置SSH免密钥登陆

**ssh-keygen** **-t** **rsa**

**ssh-copy-id** 192.168.4.251

**ssh-copy-id** 192.168.4.253

**ssh-copy-id** 192.168.4.254

  
--------------以上配置所有服务器都需要配置----------------

**安装mysql与配置主从同步**  
安装mysql数据库服务

rpm -ivh mysql57-community-**release**-el6-9.noarch.rpm

yum **install** mysql-community-**server** -y

/etc/init.d/mysqld **start**

默认密码是在

[root@db254 ~]*# grep 'temporary password' /var/log/mysqld.log*

2017-08-30T08:08:05.956867Z 1 [Note] A temporary password **is** generated **for** root@localhost: f;2YK51qXlo/

修改下默认密码吧，为了后面方便操作

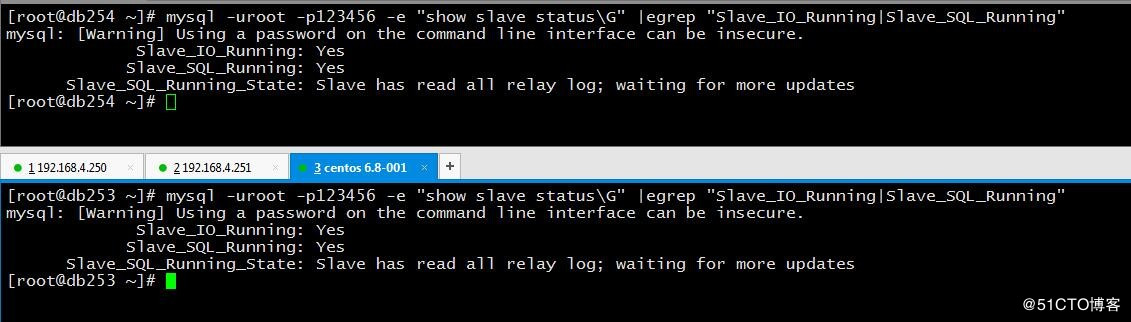
mysql> alter user 'root'@'localhost' identified by '123456';

--------------以上配置三台数据库服务器都需要配置----------------

**配置主从同步**  
Db251与db253的配置文件都需要配置开启binlog  
server\_id=251  
log\_bin=master-bin  
log\_bin\_index=master-bin.index

server\_id=253  
log\_bin=slave-bin  
log\_bin\_index=slave-bin.index  
log-slave-updates  
relay\_log\_index=slave-relay-bin.index  
relay\_log=slave-relay-bin

server\_id=254  
relay\_log\_index=slave-relay-bin.index  
relay\_log=slave-relay-bin

同步主库数据到从库，主从同步操作请参考之前的文章  
检查主从同步状态  
mysql -uroot -p123456 -e "show slave status\G" |egrep "Slave\_IO\_Running|Slave\_SQL\_Running"  
  
所有数据库节点配置MHA管理用户与权限  
grant all on *.* to mha@'192.168.4.%' identified by '123456';  
至此所有基础环境组建完成

**安装配置MHA**  
<https://downloads.mariadb.com/MHA/> 下载地址  
整个MHA服务安装过程如下：

安装依赖包

yum **install** perl-DBD-MySQL perl-Config-Tiny perl-**Log**-Dispatch -y

yum **install** perl-**Parallel**-ForkManager perl-**Time**-HiRes –y

管理节点安装

rpm -ivh http://dl.fedoraproject.org/pub/epel/6/x86\_64/epel-release-6-8.noarch.rpm

yum localinstall -y mha4mysql-manager-0.52-0.noarch.rpm

yum localinstall -y mha4mysql-node-0.52-0.noarch.rpm

其它节点安装  
yum localinstall -y mha4mysql-node-0.52-0.noarch.rpm  
安装过程中可能会报错，根据提示去解决

配置过程如下：

Mkdir /mha

[root@db250 ~]*# vim /etc/masterha\_default.cnf*

[server default]

*#刚才授权的mysql管理用戶名*

user=mha

password=123456

manager\_workdir=/mha

manager\_log=/mha/manager.log

remote\_workdir=/mha

*#ssh免密钥登录的帐号名*

ssh\_user=root

*#mysql复制帐号，用来在主从机之间同步二进制日志等*

repl\_user=slave

repl\_password=123456

*#ping间隔，用来检测master是否正常*

ping\_interval= 1

[server1]

hostname=db251

master\_binlog\_dir=/var/lib/mysql

*#候选master挂掉时候优先让它顶*

candidate\_master=1

[server2]

hostname=db253

master\_binlog\_dir=/var/lib/mysql/

candidate\_master=1

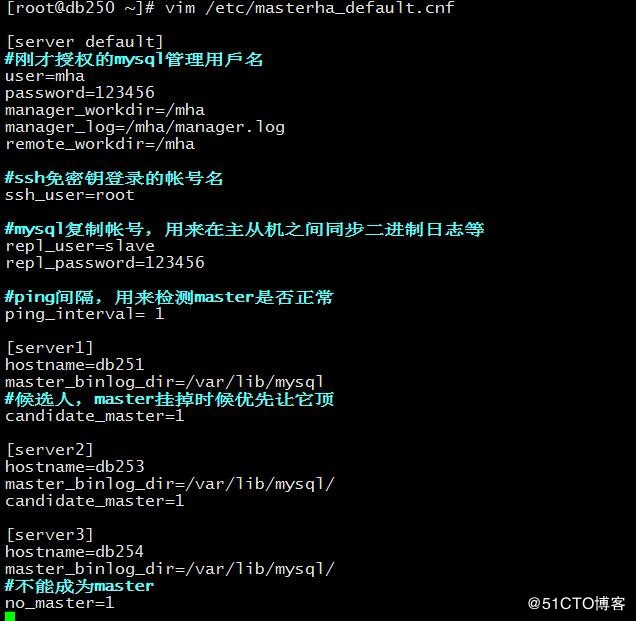
[server3]

hostname=db254

master\_binlog\_dir=/var/lib/mysql/

no\_master=1

*#不能成为master*



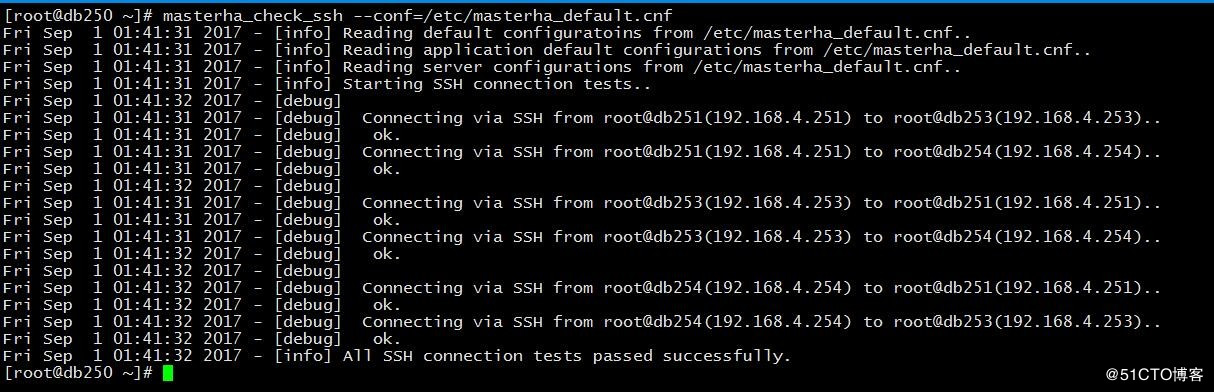
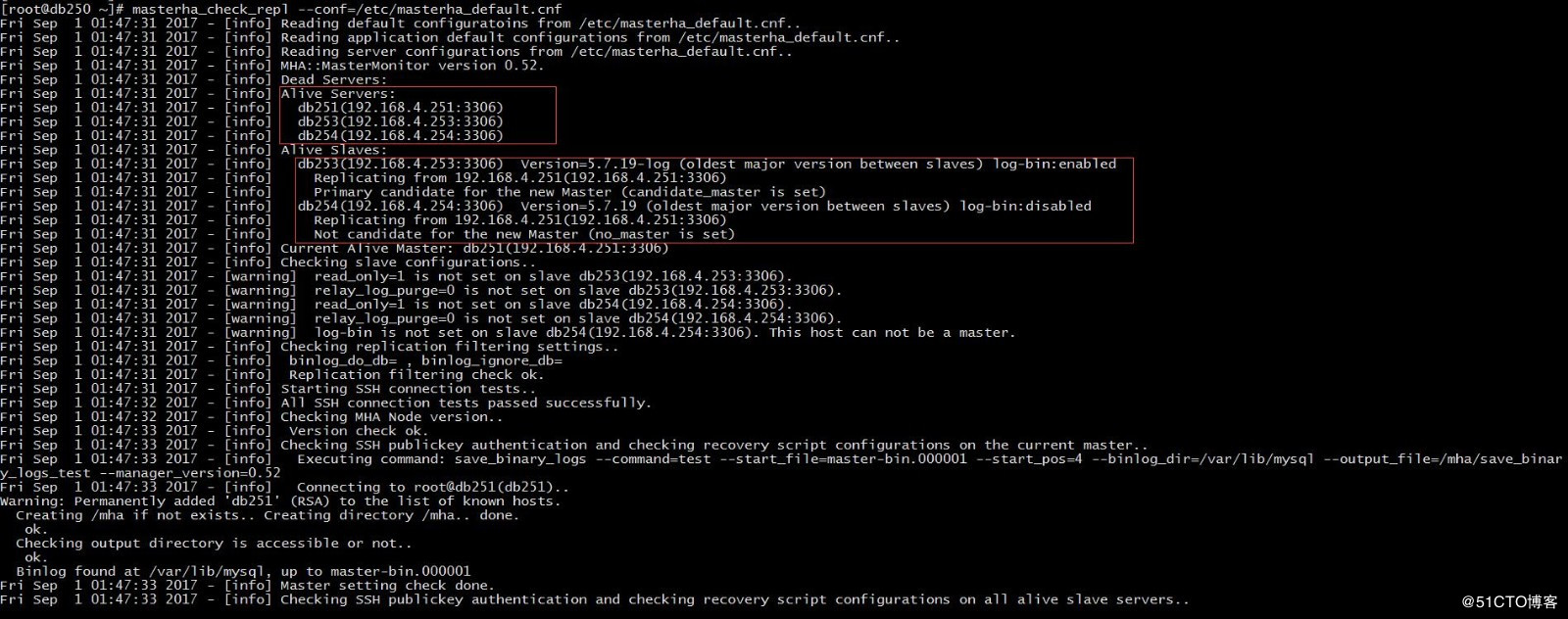
**masterha\_check\_ssh 检查MHA的SSH配置状况  
masterha\_check\_repl 检查MySQL复制状况  
masterha\_manger 启动MHA  
masterha\_check\_status 检测当前MHA运行状态  
masterha\_master\_monitor 检测master是否宕机  
masterha\_master\_switch 控制故障转移（自动或者手动）  
masterha\_conf\_host 添加或删除配置的server信息**

验证操作

[root@db250 ~]*# masterha\_check\_ssh --conf=/etc/masterha\_default.cnf*

检查主从同步

masterha\_check\_repl --conf=/etc/masterha\_default.cnf

启动MHA服务

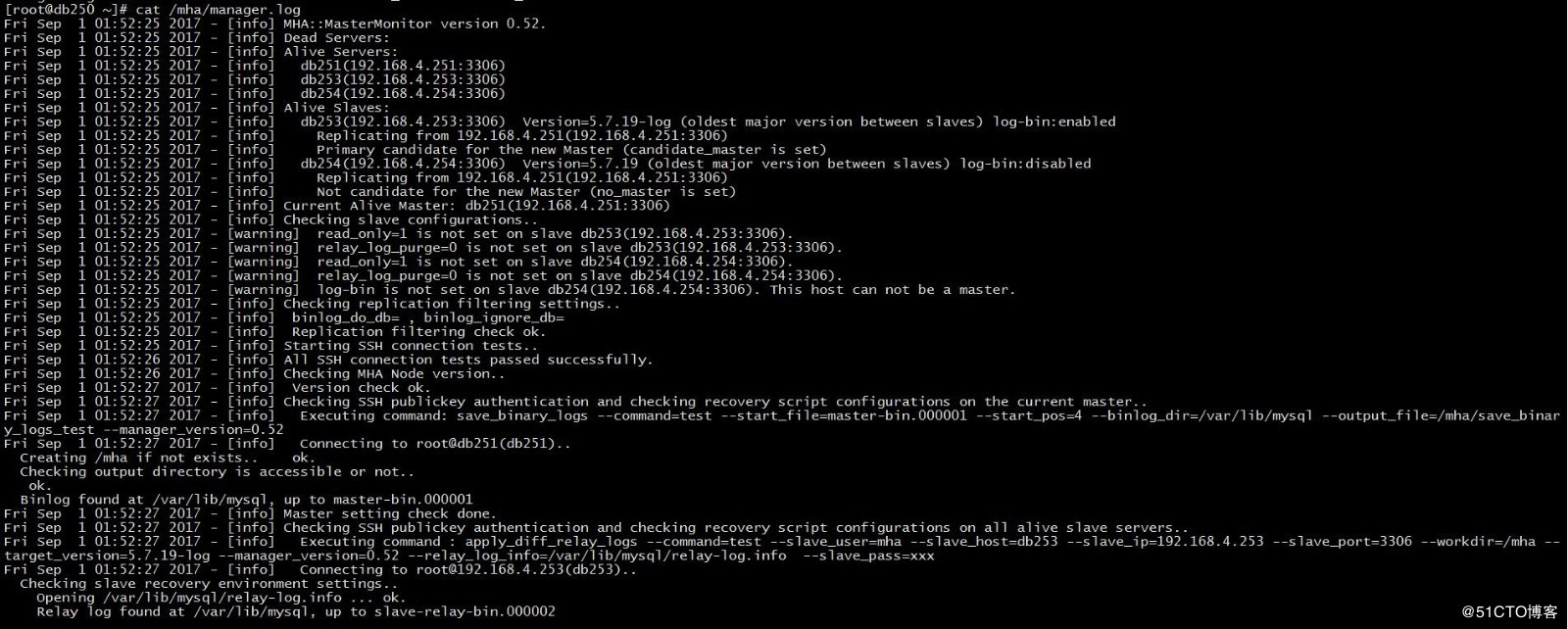
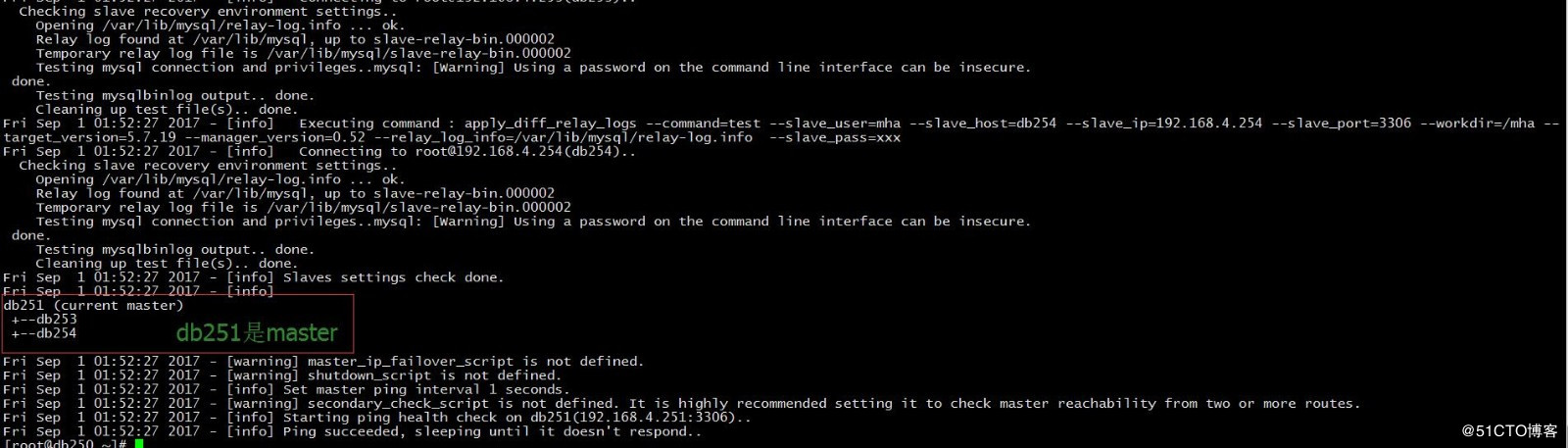
[root@db250 ~]*# masterha\_manager --conf=/etc/masterha\_default.cnf &*

[1] 2035

[root@db250 ~]*# Fri Sep 1 01:52:25 2017 - [info] Reading default configuratoins from /etc/masterha\_default.cnf..*

Fri Sep 1 01:52:25 2017 - [info] Reading application **default** configurations **from** /etc/masterha\_default.cnf..

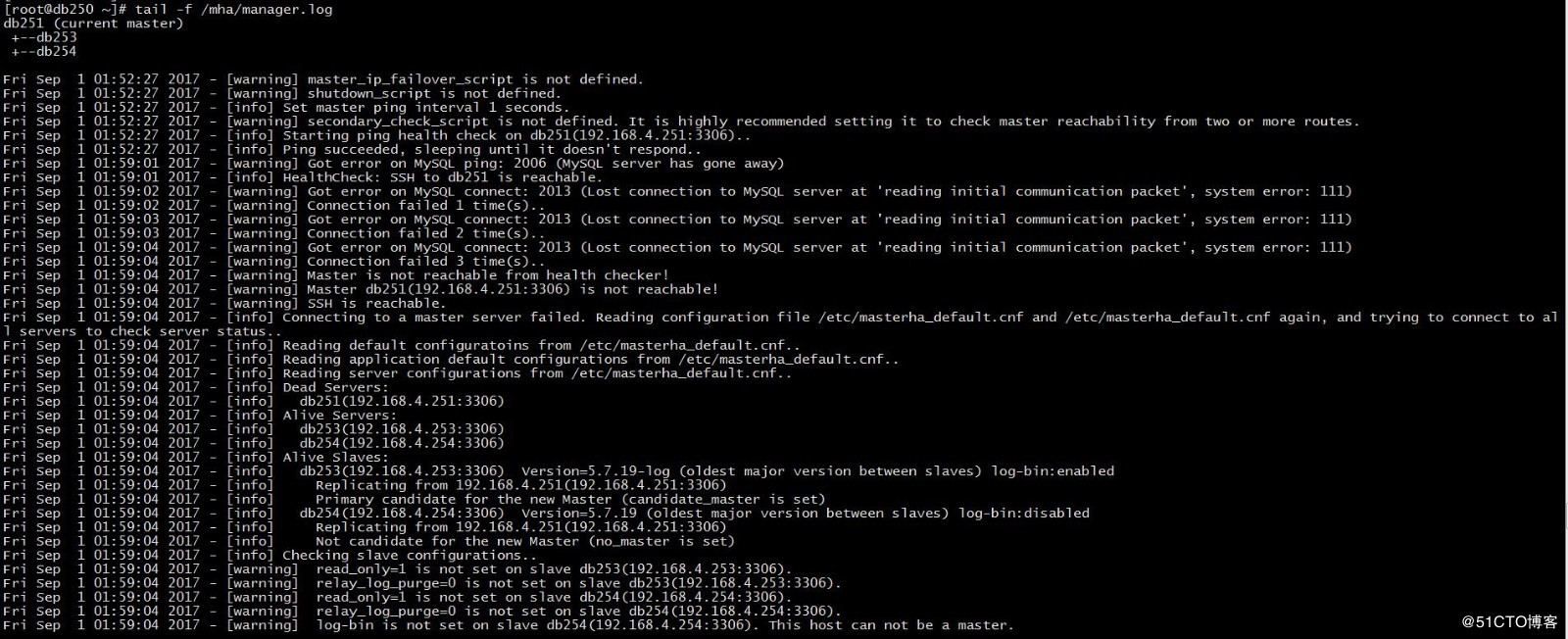
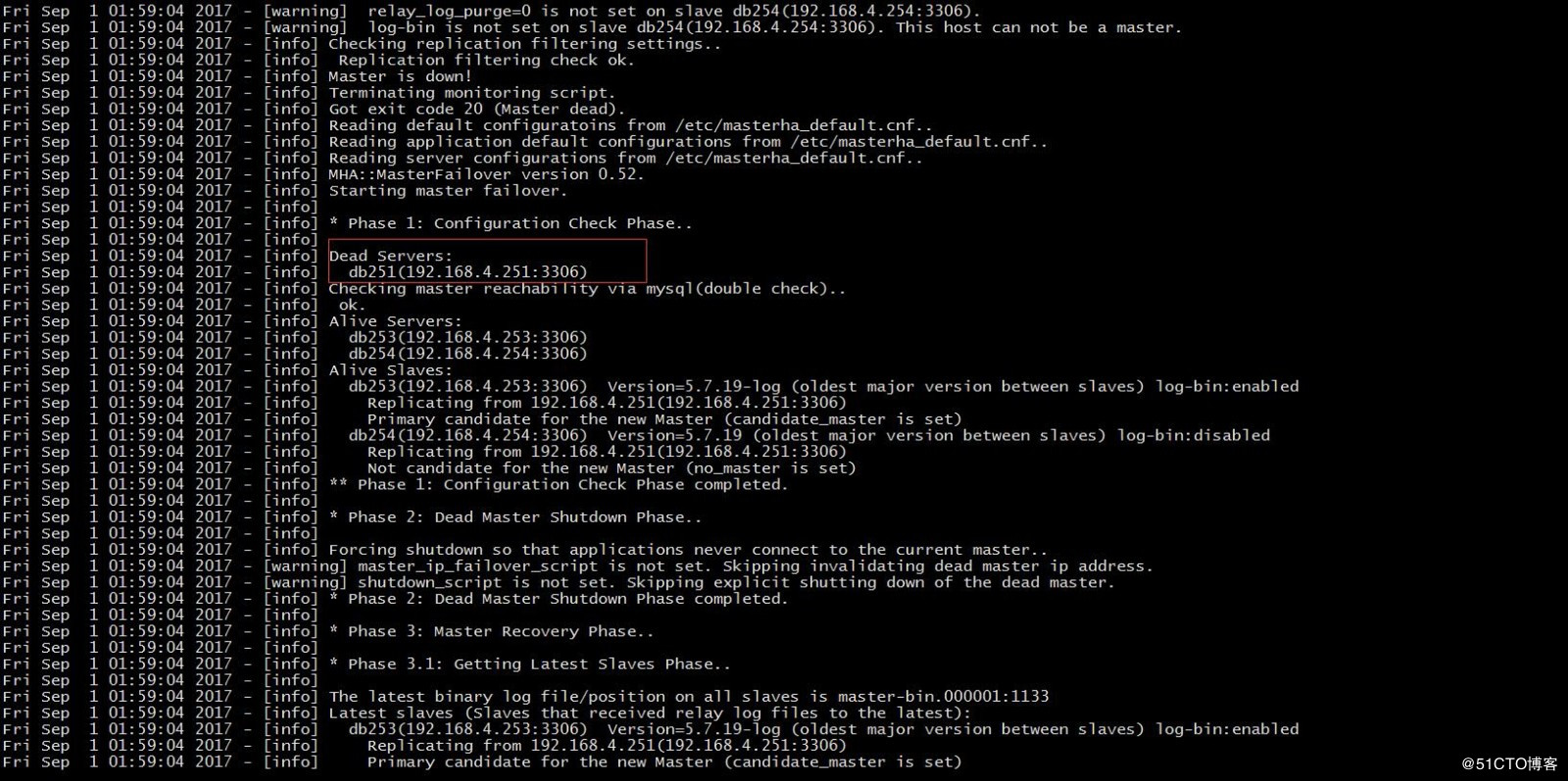
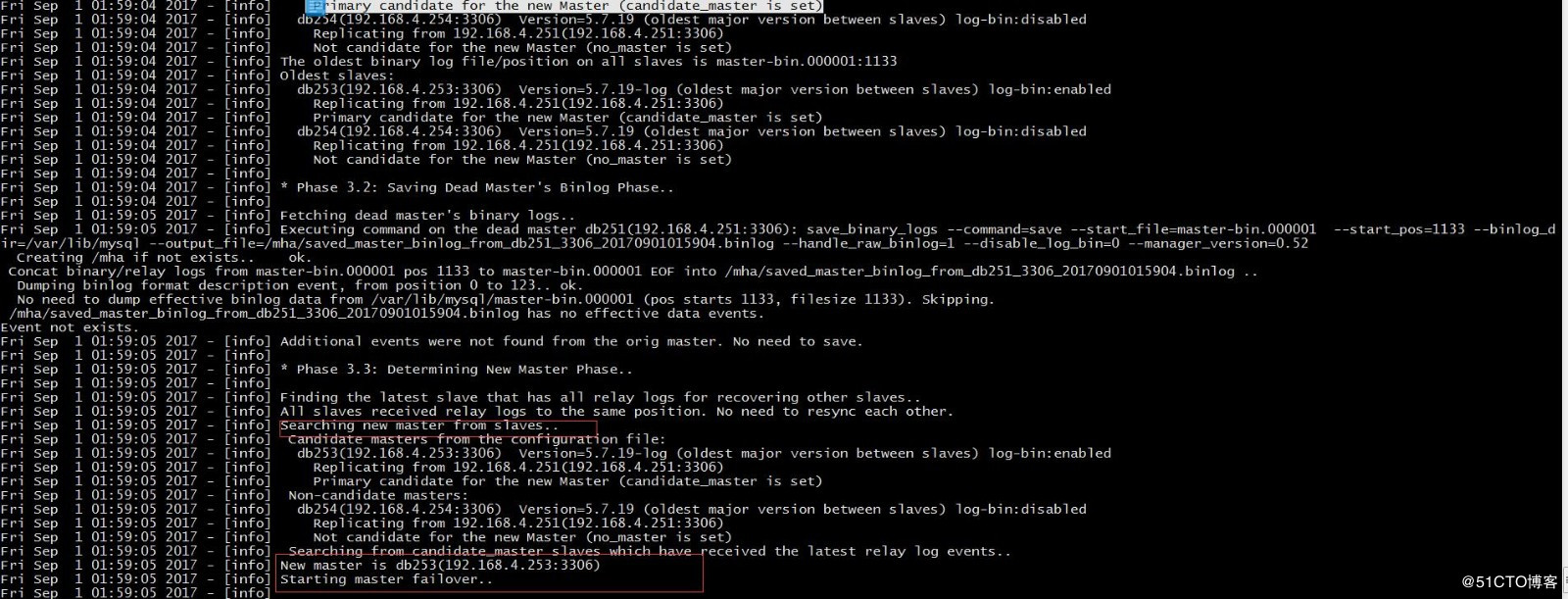
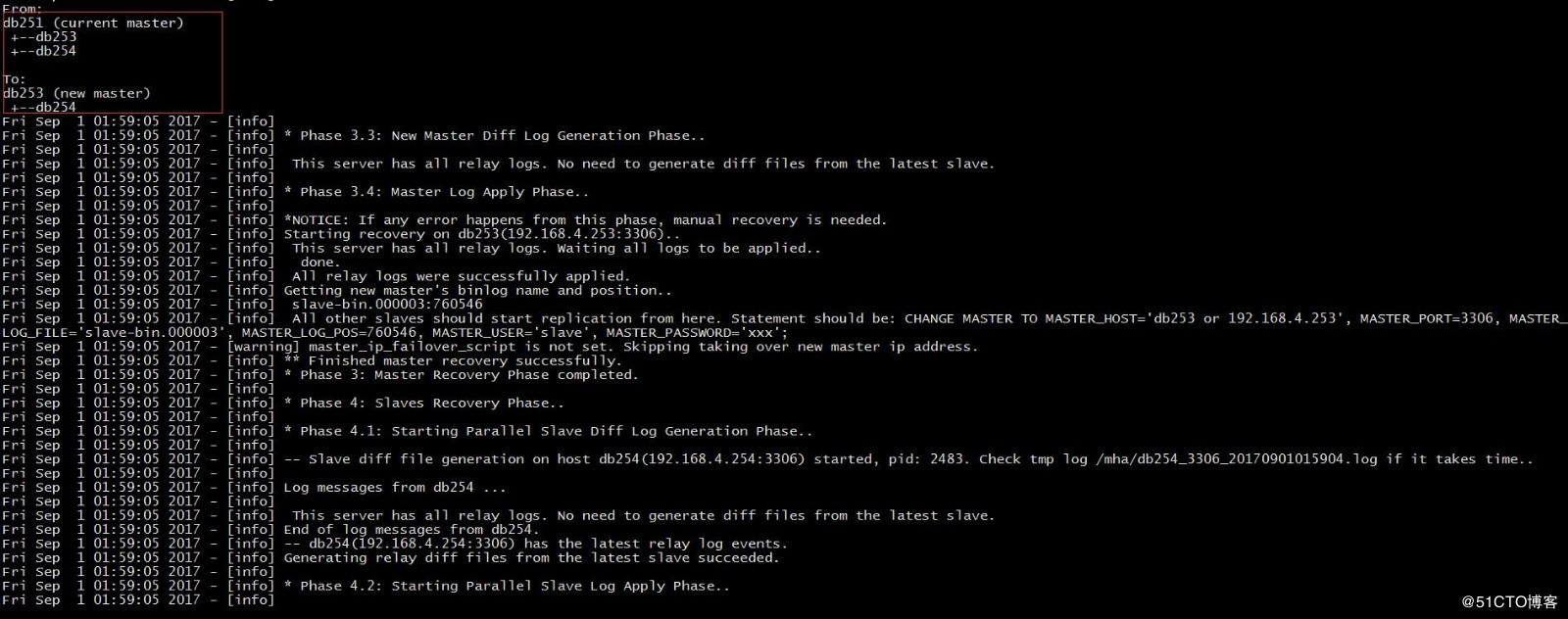
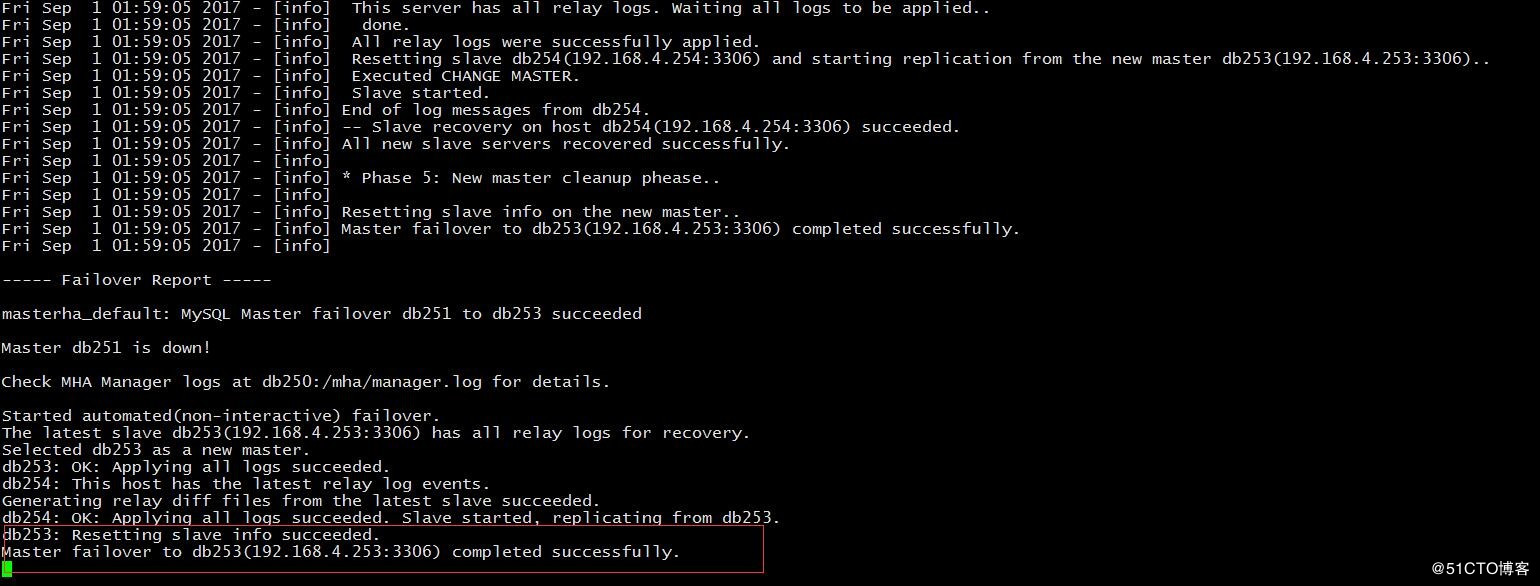
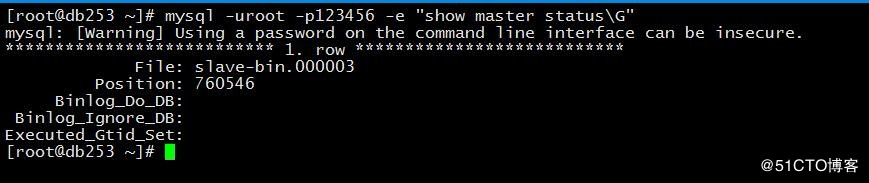
Fri Sep 1 01:52:25 2017 - [info] Reading server configurations **from** /etc/masterha\_default.cnf..

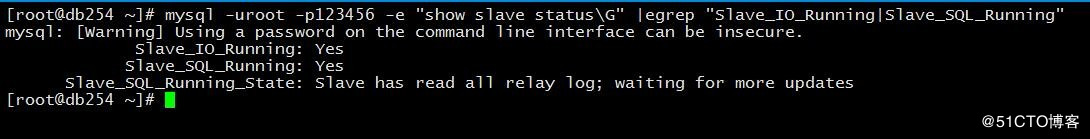
查看启动日志  
   
  
好吧，到此MHA高可用总算成功了，也算可以的哈！

接下来模拟下故障，将db251关闭

[root@db251 ~]*# /etc/init.d/mysqld stop*

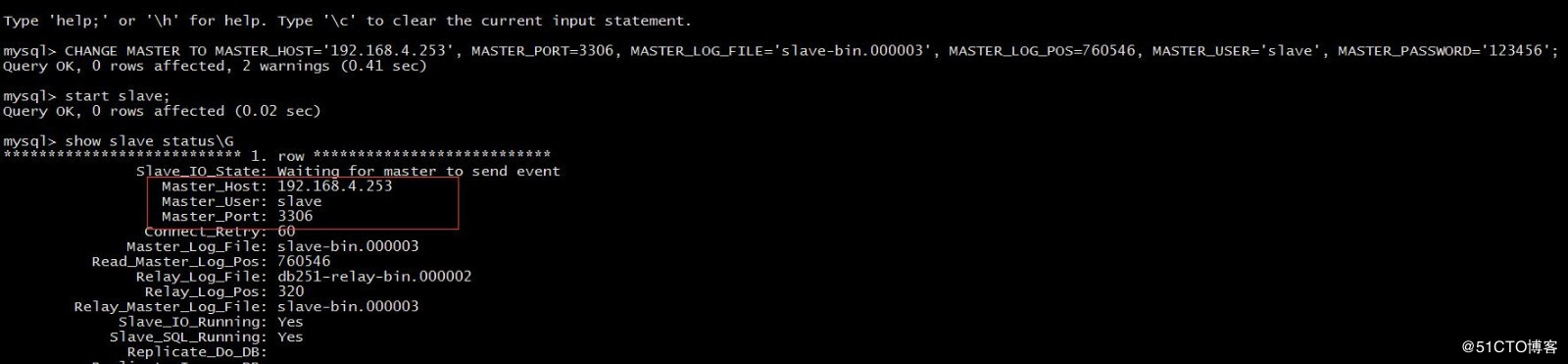
Stopping mysqld: [ OK ]

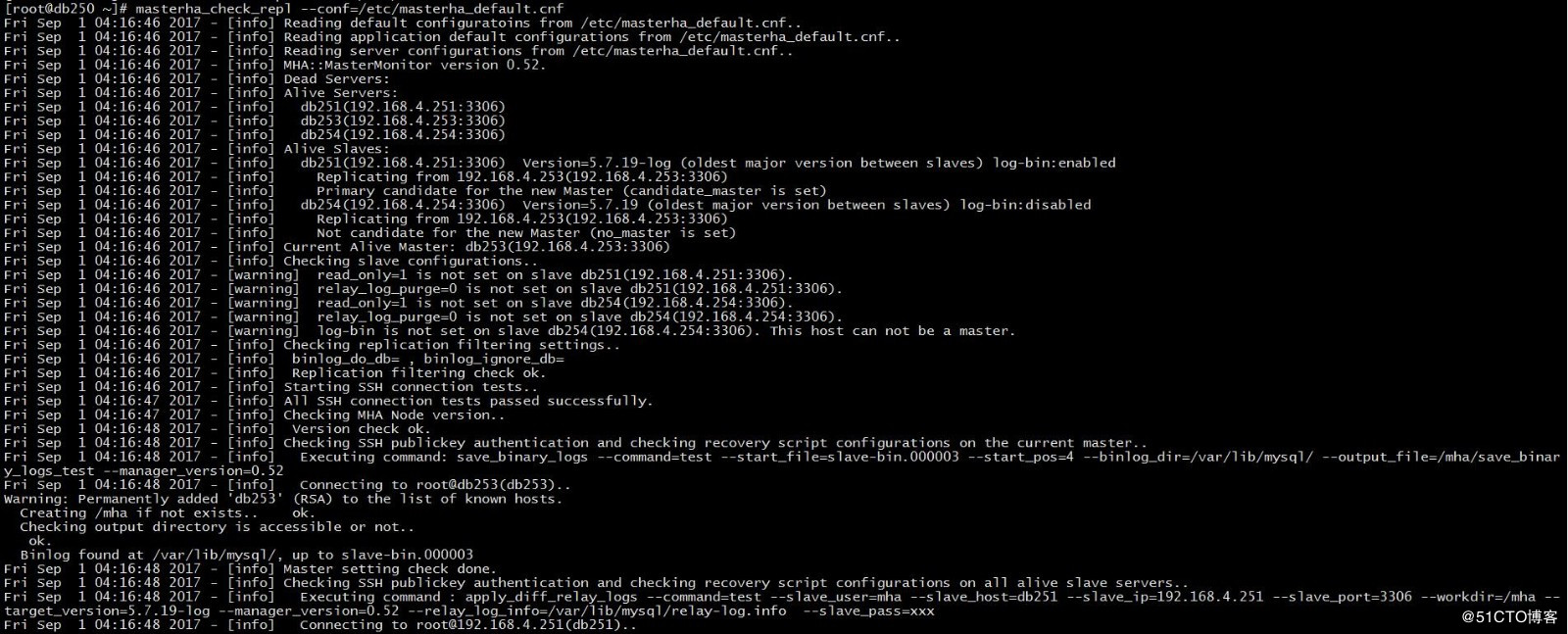
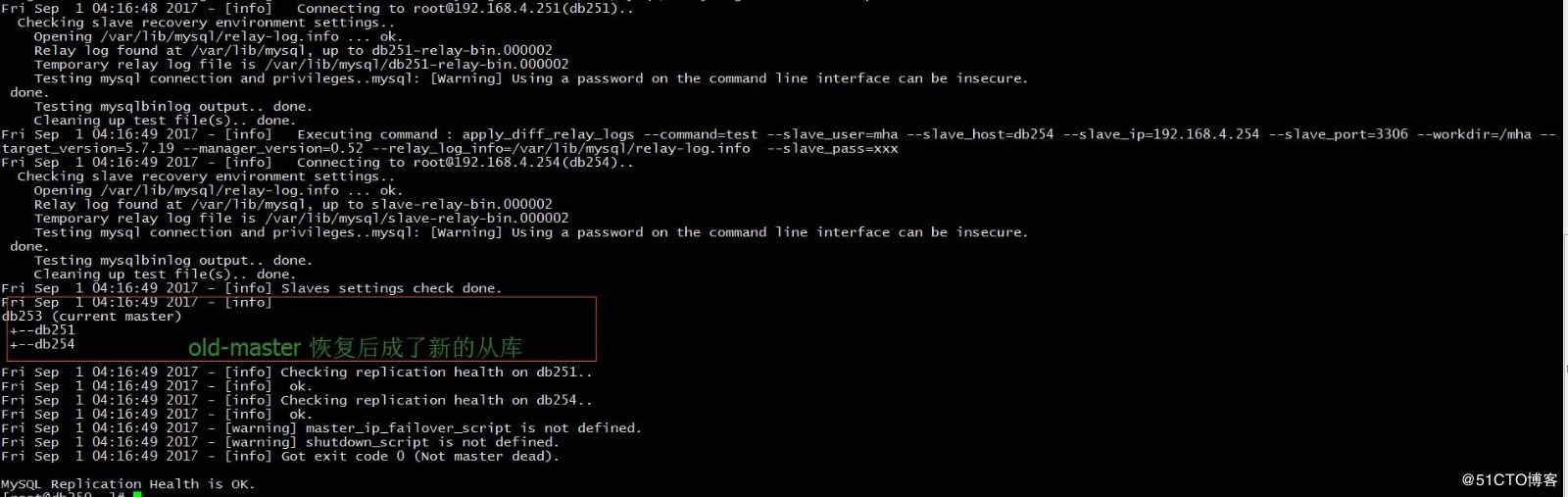
查看日志  
   
  
  
  
  
发现新的主库已切换到db253这台服务器上了，接下来查看下主从同步的情况  


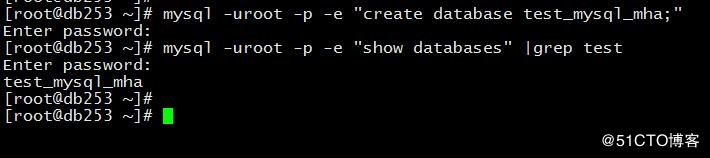
  
当原来的主服务器故障修复后，需要手工将原主变成从

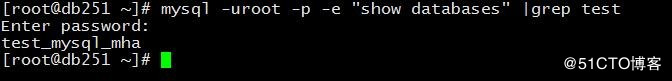
[root@db250 ~]*# grep -i "All other slaves should start" /mha/manager.log*

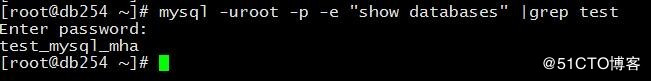
Fri Sep 1 01:59:05 2017 - [info] All other slaves should start replication **from** here. Statement should be: CHANGE MASTER TO MASTER\_HOST='db253 or 192.168.4.253', MASTER\_PORT=3306, MASTER\_LOG\_FILE='slave-bin.000003', MASTER\_LOG\_POS=760546, MASTER\_USER='slave', MASTER\_PASSWORD='xxx';

因此此时我在原主库上执行操作将修复的主库改为从库  


  
  
接下来在新的主库上写入数据来测试同步情况







版权声明：原创作品，如需转载，请注明出处。否则将追究法律责任