MHA高可用架构——主从在线切换

目 录

[MHA高可用架构——主从在线切换 1](#_Toc449106527)

[一、前言： 1](#_Toc449106528)

[1.1 MHA介绍 1](#_Toc449106529)

[1.2 MHA有如下特性： 1](#_Toc449106530)

[1.2.1. 主服务器的自动监控和故障转移 1](#_Toc449106531)

[1.2.2. 交互式主服务器故障转移 2](#_Toc449106532)

[1.2.3. 非交互式的主故障转移 2](#_Toc449106533)

[1.2.4. 在线切换主服务器 2](#_Toc449106534)

[二． MHA所需条件 2](#_Toc449106535)

[2.1. SSH公钥验证 3](#_Toc449106536)

[2.2. 操作系统 3](#_Toc449106537)

[2.3. 单台可写主服务器和多台从服务器或只读主服务器 3](#_Toc449106538)

[2.4. 在三层或三层以上复制情况下 3](#_Toc449106539)

[2.5. mysql版本5.0或更高 4](#_Toc449106540)

[2.6. mysqlbinlog版本3.3或更高 4](#_Toc449106541)

[2.7. 候选主服务器log-bin必须开启 4](#_Toc449106542)

[2.8. 所有服务器上的二进制日志和中继日志过滤规则必须相同 5](#_Toc449106543)

[2.9. 候选主服务器上的复制用户必须存在 5](#_Toc449106544)

[2.10. 保留中继日志和定期清理 5](#_Toc449106545)

[2.11. LOAD DATA INFILE不要使用基于语句型的二进制日志 6](#_Toc449106546)

[三．MHA工作流程 7](#_Toc449106547)

[3.1. 监控和故障转移过程 7](#_Toc449106548)

[3.2. 在线切换过程 8](#_Toc449106549)

[四．安装MHA操作步骤 8](#_Toc449106550)

[4.1. 环境简介 8](#_Toc449106551)

[4.2. 创建安装包目录 9](#_Toc449106552)

[4.3. 安装MHA Node 9](#_Toc449106553)

[4.4. 安装MHA Manager 9](#_Toc449106554)

[五．详细配置流程 10](#_Toc449106555)

[5.1 配置主从结构 10](#_Toc449106556)

[5.1.1 主库主配置文件 10](#_Toc449106557)

[5.1.2 从库配置主文件 13](#_Toc449106558)

[5.1.3 参数详细介绍 16](#_Toc449106559)

[5.2 主从同步操作 16](#_Toc449106560)

[5.2.1 主库创建同步帐号 16](#_Toc449106561)

[5.2.2 主库数据导入从库 16](#_Toc449106562)

[5.3 创建MySQL管理用户 17](#_Toc449106563)

[5.3.1 删除多余用户 17](#_Toc449106564)

[5.3.2 创建管理帐号 17](#_Toc449106565)

[5.4. 配置MHA工作环境 18](#_Toc449106566)

[5.4.1 创建工作目录 18](#_Toc449106567)

[5.4.2 修改配置文件 18](#_Toc449106568)

[5.4.3 主从failover脚本 19](#_Toc449106569)

[5.5. SSH免密码登录 24](#_Toc449106570)

[5.5.1. 测试SSH免密码登录 24](#_Toc449106571)

[5.5.2. 测试数据库同步 26](#_Toc449106572)

[5.5.3. 定时清理relay\_log日志 30](#_Toc449106573)

[5.6. 启动MHA 31](#_Toc449106574)

[六. 测试流程操作 31](#_Toc449106575)

[6.1 清空manager日志 31](#_Toc449106576)

[6.2 启动MHA 31](#_Toc449106577)

[6.2.1 nohup启动MHA 31](#_Toc449106578)

[6.2.2 查看启动日志 31](#_Toc449106579)

[6.3 关闭主库的MySQL 36](#_Toc449106580)

[6.3.1 杀死mysql进程 36](#_Toc449106581)

[6.3.2 查看manager日志 36](#_Toc449106582)

[6.3.3 查看从库同步的主库 45](#_Toc449106583)

[七、在线恢复主主库 47](#_Toc449106584)

[7.1 启动主主库MySQL 47](#_Toc449106585)

[7.2 将主主库同步到备主库 47](#_Toc449106586)

[7.3 恢复主主库 49](#_Toc449106587)

[7.4 查看从库同步状态 55](#_Toc449106588)

[7.5 小结： 56](#_Toc449106589)

一、前言：

1.1 MHA介绍

MHA自动化主服务器故障转移，快速将从服务器晋级为主服务器(通常在10-30s)，而不影响复制的一致性，不需要花钱买更多的新服务器，不会有性能损耗，容易安装，不必更改现有的部署环境，适用于任何存储引擎。

MHA提供在线主服务器切换，改变先正运行的主服务器到另外一台上，这个过程只需0.5-2s的时间，这个时间内数据无法写入。

MHA Manager通过ssh连接mysql slave服务器。

虽然MHA试图从挡掉的主服务器上保存二进制日志，并不是总是可行的。例如，如果主服务器硬件故障或无法通过ssh访问，MHA没法保存二进制日志，只进行故障转移而丢失最新数据。

使用半同步复制，可以大大降低数据丢失的风险。MHA可以与半同步复制结合起来。如果只有一个slave已经收到了最新的二进制日志，MHA可以将最新的二进制日志应用于其他所有的slave服务器上，因此他们彼此保持一致性。

1.2 MHA有如下特性：

1.2.1. 主服务器的自动监控和故障转移

MHA监控复制架构的主服务器，一旦检测到主服务器故障，就会自动进行故障转移。即使有些从服务器没有收到最新的relay log，MHA自动从最新的从服务器上识别差异的relay log并把这些日志应用到其他从服务器上，因此所有的从服务器保持一致性了。MHA通常在几秒内完成故障转移，9-12秒可以检测出主服务器故障，7-10秒内关闭故障的主服务器以避免脑裂，几秒中内应用差异的relay log到新的主服务器上，整个过程可以在10-30s内完成。还可以设置优先级指定其中的一台slave作为master的候选人。由于MHA在slaves之间修复一致性，因此可以将任何slave变成新的master，而不会发生一致性的问题，从而导致复制失败。

1.2.2. 交互式主服务器故障转移

可以只使用MHA的故障转移，而不用于监控主服务器，当主服务器故障时，人工调用MHA来进行故障故障。

1.2.3. 非交互式的主故障转移

不监控主服务器，但自动实现故障转移。这种特征适用于已经使用其他软件来监控主服务器状态，比如heartbeat来检测主服务器故障和虚拟IP地址接管，可以使用MHA来实现故障转移和slave服务器晋级为master服务器。

1.2.4. 在线切换主服务器

在许多情况下，需要将现有的主服务器迁移到另外一台服务器上。比如主服务器硬件故障，RAID控制卡需要重建，将主服务器移到性能更好的服务器上等等。维护主服务器引起性能下降，导致停机时间至少无法写入数据。另外，阻塞或杀掉当前运行的会话会导致主主之间数据不一致的问题发生。MHA提供快速切换和优雅的阻塞写入，这个切换过程只需要0.5-2s的时间，这段时间内数据是无法写入的。在很多情况下，0.5-2s的阻塞写入是可以接受的。因此切换主服务器不需要计划分配维护时间窗口(呵呵，不需要你在夜黑风高时通宵达旦完成切换主服务器的任务)。

**本文参考资料来源：**

<http://www.educity.cn/wenda/399310.html>

<http://www.mamicode.com/info-detail-1181862.html>

二． MHA所需条件

2.1. SSH公钥验证

MHA管理节点通过ssh连接mysql服务器，MHA节点通过scp发送最新的relay log到其他slaves服务器上。为了使这些过程自动化，使用SSH公钥验证密码。

2.2. 操作系统

MHA目前只支持Linux系统

2.3. 单台可写主服务器和多台从服务器或只读主服务器

当主服务器当掉时，MHA修复从服务器之间数据一致性。MHA试图从当掉的主服务器上保存尚未发送的二进制日志文件并应用于所有从服务器。如果只有一个从服务器，就不需在意从服务器之间一致性问题。即使只有一个从服务器，MHA也会从当掉的主服务器上保存尚未发送的二进制日志事件只要能通过ssh访问到主服务器。使用半同步复制可以解决当主服务器当掉后，无法ssh到主服务器上保存尚未发送的二进制日志事件。

从MHA Manager0.52版本开始，支持多主复制结构。只允许其中一台主服务器可写，其他主服务器必须设置read-only=1。默认情况下，被管理服务器应该是两层复制结构。

2.4. 在三层或三层以上复制情况下

默认情况下，MHA不支持三层或三层以上的复制结构。如master1—master2—-slave3。MHA故障转移和恢复是直接从从服务器中选择一台作为当前的主主服务器。MHA可以管理master1和master2，当master1当掉后，将master2作为主，MHA不会监控和恢复slave3因为slave3是从不同的主服务器上(master2)复制的。为了使MHA工作在这种架构下，需要做如下设置：

只在MHA配置文件中配置master1和master2

在MHA配置文件中所有主机上设置multi\_tier\_slave=1

在这种情况下，MHA只管理主主服务器和二层的从服务器，在故障转移过程中，三层从服务器仍然可以正常工作的。

2.5. mysql版本5.0或更高

MHA支持mysql5.0或以上版本。因为从mysql5.0版本起二进制日志格式(binlog v4格式)改变了。当MHA解析二进制日志来确定目标中继日志位置，是使用v4格式的。MySQL版本不得低于5.0.60。

2.6. mysqlbinlog版本3.3或更高

MHA在目标从服务器上应用二进制事件使用mysqlbinlog。如果主服务器使用基于行格式复制，基于行格式的事件写入到二进制文件中，这种二进制日志格式的文件只能被MySQL5.1或更高版本的mysqlbinlog解析。MySQL5.0.60以下版本中的mysqlbinlog不支持基于行格式的。

2.7. 候选主服务器log-bin必须开启

如果当前的从服务器没有开启log-bin，那么将不可能成为主服务器。MHA管理节点会检测是否有配置log-bin。如果当前所有从服务器都没有设置log-bin，那么MHA不进行故障转移。

2.8. 所有服务器上的二进制日志和中继日志过滤规则必须相同

binlog-do-db和replicate-ignore-db设置必须相同。MHA在启动时候会检测过滤规则，如果过滤规则不同，MHA不启动监控和故障转移。

2.9. 候选主服务器上的复制用户必须存在

当故障转移后，所有从服务器上将执行change master to命令。

2.10. 保留中继日志和定期清理

默认情况下，从服务器上的中继日志在SQL线程执行完后会被自动删除的。但是这些中继日志在恢复其他从服务器时候可能会被用到，因此需要禁用中继日志的自动清除和定期清除旧的中继日志。定期清除中继日志需要考虑到复制延时的问题。在ext3文件系统下，删除大的文件需要一定的时间，会导致严重的复制延时。为了避免复制延时，暂时为中继日志创建硬链接。

MHA节点包含pure\_relay\_logs命令工具，它可以为中继日志创建硬链接，执行SET GLOBAL relay\_log\_purge=1，等待几秒中以便SQL线程切换到新的中继日志，再执行SET GLOBAL relay\_log\_purge=0。

pure\_relay\_logs参数如下所示：

–user mysql用户名

–password mysql密码

–host mysql服务器地址

–port 端口号

–workdir 创建和删除中继日志硬链接目录。成功执行脚本后，硬链接的中继日志文件将被删除。默认目录是/var/tmp。

–disable\_relay\_log\_purge 如果relay\_log\_purge=1，purge\_relay\_logs脚本将退出不做任何事情。设置–disable\_relay\_log\_purge参数，purge\_relay\_logs脚本不会退去，且自动设置relay\_log\_purge=0。

定期执行purge\_relay\_logs脚本：

Purge\_relay\_logs脚本删除中继日志不会阻塞SQL线程。因此在每台从服务器上设置计划任务定期清除中继日志。

00 00 \* \* \* /usr/bin/purge\_relay\_logs –user=root –password=passwd –disable\_relay\_log\_purge $amp;>amp;$gt; /data/masterha/log/purge\_relay\_logs.log 2$amp;>amp;$amp;1

最好在每台从服务器上不同时间点执行计划任务。

2.11. LOAD DATA INFILE不要使用基于语句型的二进制日志

如果使用非事务性存储引擎，在执行完LOAD DATA INFILE基于语句型二进制日志时，主服务器当掉，MHA可能不会产生差异的中继日志事件。使用LOAD DATA INFILE基于语句型二进制日志有一些已知问题，在mysql5.1版本中不建议使用，同时还会引起严重的复制延时，因此没有理由使用它。

三．MHA工作流程

3.1. 监控和故障转移过程

检测复制设置和确定当前主服务器

监控主服务器

检测主服务器当掉

再次检测从服务器配置

关闭当掉的主服务器(可选)

恢复一个新的主服务器

激活新的主服务器

恢复其余的从服务器

告警(可选)

3.2. 在线切换过程

检测复制设置和确定当前主服务器

确定新的主服务器

阻塞写入到当前主服务器

等待所有从服务器赶上复制

授予写入到新的主服务器

重新设置从服务器

四．安装MHA操作步骤

MHA节点包含三个脚本，依赖perl模块。

save\_binary\_logs:保存和复制当掉的主服务器二进制日志

apply\_diff\_relay\_logs:识别差异的relay log事件，并应用于其他salve服务器

purge\_relay\_logs:清除relay log文件

需要在所有mysql服务器上安装MHA节点，MHA管理服务器也需要安装。MHA管理节点模块内部依赖MHA节点模块。MHA管理节点通过ssh连接管理mysql服务器和执行MHA节点脚本。MHA节点依赖perl的DBD::mysql模块。

4.1. 环境简介

|  |  |  |  |
| --- | --- | --- | --- |
| **主机名** | **IP地址** | **说明** | **用途** |
| master-server | 192.168.10.130 | 主库 | 主主库 |
| slave01 | 192.168.10.131 | 从库 | 备主库 |
| slave02 | 192.168.10.132 | 从库 | 从库 |
| slave03 | 192.168.10.133 | 从库 | 管理监控 |

4.2. 创建安装包目录

mkdir tools -p

cd tools/

rz mha4mysql-node-0.56-0.el6.noarch.rpm

rz mha4mysql-manager-0.56-0.el6.noarch.rpm

4.3. 安装MHA Node

在所有的mysql服务器上安装

yum install -y perl-DBD-MySQL

rpm -ihv mha4mysql-node-0.56-0.el6.noarch.rpm

4.4. 安装MHA Manager

yum -y install Parallel-ForkManager perl-Log-Dispatch perl-Time-HiRes perl-Mail-Sender perl-Mail-Sendmail perl-DBD-MySQL perl-Config-Tiny perl-Log-Dispatch perl-Config-IniFiles

rpm -ihv mha4mysql-node-0.56-0.el6.noarch.rpm

rpm -ivh mha4mysql-manager-0.56-0.el6.noarch.rpm

五．详细配置流程

5.1 配置主从结构

5.1.1 主库主配置文件

[root@master-server ~]# cat /meishi/data/meishi/my.cnf

[client]

port = 3306

socket = /tmp/mysql.sock

[mysql]

no-auto-rehash

[mysqld]

user = mysql

port = 3306

socket = /tmp/mysql.sock

basedir = /usr/local/mysql

**datadir = /meishi/data/meishi**

open\_files\_limit = 1024

back\_log = 600

max\_connections = 800

max\_connect\_errors = 3000

table\_cache = 614

external-locking = FALSE

max\_allowed\_packet =8M

sort\_buffer\_size = 1M

join\_buffer\_size = 1M

thread\_cache\_size = 100

thread\_concurrency = 2

query\_cache\_size = 2M

query\_cache\_limit = 1M

query\_cache\_min\_res\_unit = 2k

#default\_table\_type = InnoDB

thread\_stack = 192K

#transaction\_isolation = READ-COMMITTED

tmp\_table\_size = 2M

max\_heap\_table\_size = 2M

long\_query\_time = 1

#log\_long\_format

#log-error = /meishi/data/meishi/error.log

#log-slow-queries = /meishi/data/meishi/slow.log

pid-file = /meishi/data/meishi/mysql.pid

**log-bin = /meishi/data/meishi/mysql-bin**

**relay-log = /meishi/data/meishi/relay-bin**

relay-log-info-file = /meishi/data/meishi/relay-log.info

binlog\_cache\_size = 1M

max\_binlog\_cache\_size = 1M

max\_binlog\_size = 2M

expire\_logs\_days = 7

key\_buffer\_size = 16M

read\_buffer\_size = 1M

read\_rnd\_buffer\_size = 1M

bulk\_insert\_buffer\_size = 1M

#myisam\_sort\_buffer\_size = 1M

#myisam\_max\_sort\_file\_size = 10G

#myisam\_max\_extra\_sort\_file\_size = 10G

#myisam\_repair\_threads = 1

#myisam\_recover

lower\_case\_table\_names = 1

skip-name-resolve

slave-skip-errors = 1032,1062

replicate-ignore-db=mysql

**server-id = 1**

innodb\_additional\_mem\_pool\_size = 4M

innodb\_buffer\_pool\_size = 32M

innodb\_data\_file\_path = ibdata1:128M:autoextend

innodb\_file\_io\_threads = 4

innodb\_thread\_concurrency = 8

innodb\_flush\_log\_at\_trx\_commit = 2

innodb\_log\_buffer\_size = 2M

innodb\_log\_file\_size = 4M

innodb\_log\_files\_in\_group = 3

innodb\_max\_dirty\_pages\_pct = 90

innodb\_lock\_wait\_timeout = 120

innodb\_file\_per\_table = 0

[mysqldump]

quick

max\_allowed\_packet = 2M

[mysqld\_safe]

log-error=/meishi/data/meishi/mysql.err

pid-file=/meishi/data/meishi/mysqld.pid

5.1.2 从库配置主文件

[root@slave01 ~]# cat /meishi/data/meishi/my.cnf

[client]

port = 3306

socket = /tmp/mysql.sock

[mysql]

no-auto-rehash

[mysqld]

user = mysql

port = 3306

socket = /tmp/mysql.sock

basedir = /usr/local/mysql

**datadir = /meishi/data/meishi**

open\_files\_limit = 1024

back\_log = 600

max\_connections = 800

max\_connect\_errors = 3000

table\_cache = 614

**read\_only = 1**

**relay\_log\_purge = 0**

external-locking = FALSE

max\_allowed\_packet =8M

sort\_buffer\_size = 1M

join\_buffer\_size = 1M

thread\_cache\_size = 100

thread\_concurrency = 2

query\_cache\_size = 2M

query\_cache\_limit = 1M

query\_cache\_min\_res\_unit = 2k

#default\_table\_type = InnoDB

thread\_stack = 192K

#transaction\_isolation = READ-COMMITTED

tmp\_table\_size = 2M

max\_heap\_table\_size = 2M

long\_query\_time = 1

#log\_long\_format

#log-error = /meishi/data/meishi/error.log

#log-slow-queries = /meishi/data/meishi/slow.log

pid-file = /meishi/data/meishi/mysql.pid

**log-bin = /meishi/data/meishi/mysql-bin**

**relay-log = /meishi/data/meishi/relay-bin**

relay-log-info-file = /meishi/data/meishi/relay-log.info

binlog\_cache\_size = 1M

max\_binlog\_cache\_size = 1M

max\_binlog\_size = 2M

expire\_logs\_days = 7

key\_buffer\_size = 16M

read\_buffer\_size = 1M

read\_rnd\_buffer\_size = 1M

bulk\_insert\_buffer\_size = 1M

#myisam\_sort\_buffer\_size = 1M

#myisam\_max\_sort\_file\_size = 10G

#myisam\_max\_extra\_sort\_file\_size = 10G

#myisam\_repair\_threads = 1

#myisam\_recover

lower\_case\_table\_names = 1

skip-name-resolve

slave-skip-errors = 1032,1062

replicate-ignore-db=mysql

**server-id = 2**

innodb\_additional\_mem\_pool\_size = 4M

innodb\_buffer\_pool\_size = 32M

innodb\_data\_file\_path = ibdata1:128M:autoextend

innodb\_file\_io\_threads = 4

innodb\_thread\_concurrency = 8

innodb\_flush\_log\_at\_trx\_commit = 2

innodb\_log\_buffer\_size = 2M

innodb\_log\_file\_size = 4M

innodb\_log\_files\_in\_group = 3

innodb\_max\_dirty\_pages\_pct = 90

innodb\_lock\_wait\_timeout = 120

innodb\_file\_per\_table = 0

[mysqldump]

quick

max\_allowed\_packet = 2M

[mysqld\_safe]

log-error=/meishi/data/meishi/mysql.err

pid-file=/meishi/data/meishi/mysqld.pid

5.1.3 参数详细介绍

**1）各从库应设置relay\_log\_purge=0**

否则收到以下告警信息mysql -e 'set global relay\_log\_purge=0' 动态修改该参数，因为随时slave会提升为master。

**2）各从库设置read\_only=1**

否则收到以下告警信息 mysql -e 'set global read\_only=1' 动态修改该参数，因为随时slave会提升为master

5.2 主从同步操作

5.2.1 主库创建同步帐号

mysql> grant replication slave on \*.\* to ’rep’@’192.168.10.%’ identified by ’rep’;

Query OK, 0 rows affected (0.01 sec)

5.2.2 主库数据导入从库

**主库操作**

mysqldump -uroot -pmeishi -A -F -x -R --events --master-data=2 >/opt/rep.sql

scp -rp /opt/rep.sql [root@'192.168.10.131':/opt](mailto:root@'192.168.10.131':/opt)

scp -rp /opt/rep.sql [root@'192.168.10.132':/opt](mailto:root@'192.168.10.132':/opt)

scp -rp /opt/rep.sql [root@'192.168.10.133':/opt](mailto:root@'192.168.10.133':/opt)

**从库操作**

mysql -uroot -pmeishi </opt/rep.sql

CHANGE MASTER TO

MASTER\_HOST='192.168.10.130',

MASTER\_PORT=3306,

MASTER\_USER='rep',

MASTER\_PASSWORD='rep';

mysql –uroot –pmeishi

start slave;

show slave status\G

5.3 创建MySQL管理用户

5.3.1 删除多余用户

mysql> drop user root@'localhost';

mysql> select user,host from mysql.user;

+------+--------------+

| user | host |

+------+--------------+

| root | 127.0.0.1 |

| rep | 192.168.10.% |

+------+--------------+

3 rows in set (0.00 sec

5.3.2 创建管理帐号

mysql> grant all on \*.\* to root@'192.168.10.%' identified by 'meishi';

Query OK, 0 rows affected (0.01 sec)

**原因：之前测试MHA主从同步报错，就是数据库授权管理问题，不是使用的localhost进行管理数据库，而是使用manager管理内网网段进行检查同步。**

5.4. 配置MHA工作环境

5.4.1 创建工作目录

mkdir -p /masterha/{app1,scripts}

cp -arp /usr/bin/masterha\_\* /etc/mha/scripts/

ln -s /usr/local/mysql/bin/mysql /usr/bin/mysql

ln -s /usr/local/mysql/bin/mysqlbinlog /usr/bin/mysqlbinlog

5.4.2 修改配置文件

[root@manager ~]# cat /etc/mha/app1/app1.cnf

[server default]

user=root

password=meishi

ssh\_user=root

repl\_user=rep

repl\_password=rep

ping\_interval=1

manager\_workdir=/var/log/mha/app1

manager\_log=/var/log/mha/app1/manager.log

master\_ip\_failover\_script="/etc/mha/scripts/master\_ip\_failover"

[server1]

hostname=192.168.10.130

master\_binlog\_dir="/meishi/data/meishi"

candidate\_master=1

[server2]

hostname=192.168.10.131

master\_binlog\_dir="/meishi/data/meishi"

candidate\_master=1

[server3]

hostname=192.168.10.132

master\_binlog\_dir="/meishi/data/meishi"

[server4]

hostname=192.168.10.133

master\_binlog\_dir="/meishi/data/meishi"

no\_master=1

5.4.3 主从failover脚本

[root@manager ~]# cat /etc/mha/scripts/master\_ip\_failover

#!/usr/bin/env perl

# Copyright (C) 2011 DeNA Co.,Ltd.

#

# This program is free software; you can redistribute it and/or modify

# it under the terms of the GNU General Public License as published by

# the Free Software Foundation; either version 2 of the License, or

# (at your option) any later version.

#

# This program is distributed in the hope that it will be useful,

# but WITHOUT ANY WARRANTY; without even the implied warranty of

# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the

# GNU General Public License for more details.

#

# You should have received a copy of the GNU General Public License

# along with this program; if not, write to the Free Software

# Foundation, Inc.,

# 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

## Note: This is a sample script and is not complete. Modify the script based on your environment.

use strict;

use warnings FATAL => 'all';

use Getopt::Long;

use MHA::DBHelper;

my (

$command, $ssh\_user, $orig\_master\_host,

$orig\_master\_ip, $orig\_master\_port, $new\_master\_host,

$new\_master\_ip, $new\_master\_port, $new\_master\_user,

$new\_master\_password

);

GetOptions(

'command=s' => \$command,

'ssh\_user=s' => \$ssh\_user,

'orig\_master\_host=s' => \$orig\_master\_host,

'orig\_master\_ip=s' => \$orig\_master\_ip,

'orig\_master\_port=i' => \$orig\_master\_port,

'new\_master\_host=s' => \$new\_master\_host,

'new\_master\_ip=s' => \$new\_master\_ip,

'new\_master\_port=i' => \$new\_master\_port,

'new\_master\_user=s' => \$new\_master\_user,

'new\_master\_password=s' => \$new\_master\_password,

);

exit &main();

sub main {

if ( $command eq "stop" || $command eq "stopssh" ) {

# $orig\_master\_host, $orig\_master\_ip, $orig\_master\_port are passed.

# If you manage master ip address at global catalog database,

# invalidate orig\_master\_ip here.

my $exit\_code = 1;

eval {

# updating global catalog, etc

$exit\_code = 0;

};

if ($@) {

warn "Got Error: $@\n";

exit $exit\_code;

}

exit $exit\_code;

}

elsif ( $command eq "start" ) {

# all arguments are passed.

# If you manage master ip address at global catalog database,

# activate new\_master\_ip here.

# You can also grant write access (create user, set read\_only=0, etc) here.

my $exit\_code = 10;

eval {

my $new\_master\_handler = new MHA::DBHelper();

# args: hostname, port, user, password, raise\_error\_or\_not

$new\_master\_handler->connect( $new\_master\_ip, $new\_master\_port,

$new\_master\_user, $new\_master\_password, 1 );

## Set read\_only=0 on the new master

$new\_master\_handler->disable\_log\_bin\_local();

print "Set read\_only=0 on the new master.\n";

$new\_master\_handler->disable\_read\_only();

## Creating an app user on the new master

print "Creating app user on the new master..\n";

FIXME\_xxx\_create\_user( $new\_master\_handler->{dbh} );

$new\_master\_handler->enable\_log\_bin\_local();

$new\_master\_handler->disconnect();

## Update master ip on the catalog database, etc FIXME\_xxx;

$exit\_code = 0;

};

if ($@) {

warn $@;

# If you want to continue failover, exit 10.

exit $exit\_code;

}

exit $exit\_code;

}

elsif ( $command eq "status" ) {

# do nothing

exit 0;

}

else {

&usage();

exit 1;

}

}

sub usage {

print

"Usage: master\_ip\_failover --command=start|stop|stopssh|status --orig\_master\_host=host --orig\_master\_ip=ip --orig\_master\_port=port --new\_master\_host=host --new\_master\_ip=ip --new\_master\_port=port\n";

}

5.5. SSH免密码登录

在主库上操作

ssh-copy-id -i .ssh/id\_rsa.pub root@192.168.10.131

ssh-copy-id -i .ssh/id\_rsa.pub root@192.168.10.132

ssh-copy-id -i .ssh/id\_rsa.pub [root@192.168.10.133](mailto:root@192.168.10.133)

以此类推循环操作免密码

5.5.1. 测试SSH免密码登录

[root@manager ~]# /etc/mha/scripts/masterha\_check\_ssh --conf=/etc/mha/app1/app1.cnf

Wed Apr 20 05:11:41 2016 - [warning] Global configuration file /etc/masterha\_default.cnf not found. Skipping.

Wed Apr 20 05:11:41 2016 - [info] Reading application default configuration from /etc/mha/app1/app1.cnf..

Wed Apr 20 05:11:41 2016 - [info] Reading server configuration from /etc/mha/app1/app1.cnf..

Wed Apr 20 05:11:41 2016 - [info] Starting SSH connection tests..

Wed Apr 20 05:11:42 2016 - [debug]

Wed Apr 20 05:11:41 2016 - [debug] Connecting via SSH from root@192.168.10.130(192.168.10.130:22) to root@192.168.10.131(192.168.10.131:22)..

Wed Apr 20 05:11:41 2016 - [debug] ok.

Wed Apr 20 05:11:41 2016 - [debug] Connecting via SSH from root@192.168.10.130(192.168.10.130:22) to root@192.168.10.132(192.168.10.132:22)..

Wed Apr 20 05:11:41 2016 - [debug] ok.

Wed Apr 20 05:11:41 2016 - [debug] Connecting via SSH from root@192.168.10.130(192.168.10.130:22) to root@192.168.10.133(192.168.10.133:22)..

Wed Apr 20 05:11:41 2016 - [debug] ok.

Wed Apr 20 05:11:42 2016 - [debug]

Wed Apr 20 05:11:41 2016 - [debug] Connecting via SSH from root@192.168.10.131(192.168.10.131:22) to root@192.168.10.130(192.168.10.130:22)..

Wed Apr 20 05:11:42 2016 - [debug] ok.

Wed Apr 20 05:11:42 2016 - [debug] Connecting via SSH from root@192.168.10.131(192.168.10.131:22) to root@192.168.10.132(192.168.10.132:22)..

Wed Apr 20 05:11:42 2016 - [debug] ok.

Wed Apr 20 05:11:42 2016 - [debug] Connecting via SSH from root@192.168.10.131(192.168.10.131:22) to root@192.168.10.133(192.168.10.133:22)..

Wed Apr 20 05:11:42 2016 - [debug] ok.

Wed Apr 20 05:11:43 2016 - [debug]

Wed Apr 20 05:11:42 2016 - [debug] Connecting via SSH from root@192.168.10.132(192.168.10.132:22) to root@192.168.10.130(192.168.10.130:22)..

Wed Apr 20 05:11:42 2016 - [debug] ok.

Wed Apr 20 05:11:42 2016 - [debug] Connecting via SSH from root@192.168.10.132(192.168.10.132:22) to root@192.168.10.131(192.168.10.131:22)..

Wed Apr 20 05:11:42 2016 - [debug] ok.

Wed Apr 20 05:11:42 2016 - [debug] Connecting via SSH from root@192.168.10.132(192.168.10.132:22) to root@192.168.10.133(192.168.10.133:22)..

Wed Apr 20 05:11:42 2016 - [debug] ok.

Wed Apr 20 05:11:43 2016 - [debug]

Wed Apr 20 05:11:42 2016 - [debug] Connecting via SSH from root@192.168.10.133(192.168.10.133:22) to root@192.168.10.130(192.168.10.130:22)..

Wed Apr 20 05:11:42 2016 - [debug] ok.

Wed Apr 20 05:11:42 2016 - [debug] Connecting via SSH from root@192.168.10.133(192.168.10.133:22) to root@192.168.10.131(192.168.10.131:22)..

Wed Apr 20 05:11:43 2016 - [debug] ok.

Wed Apr 20 05:11:43 2016 - [debug] Connecting via SSH from root@192.168.10.133(192.168.10.133:22) to root@192.168.10.132(192.168.10.132:22)..

Wed Apr 20 05:11:43 2016 - [debug] ok.

Wed Apr 20 05:11:43 2016 - [info] All SSH connection tests passed successfully.

5.5.2. 测试数据库同步

[root@manager ~]# /etc/mha/scripts/masterha\_check\_repl --conf=/etc/mha/app1/app1.cnf

Wed Apr 20 05:13:26 2016 - [warning] Global configuration file /etc/masterha\_default.cnf not found. Skipping.

Wed Apr 20 05:13:26 2016 - [info] Reading application default configuration from /etc/mha/app1/app1.cnf..

Wed Apr 20 05:13:26 2016 - [info] Reading server configuration from /etc/mha/app1/app1.cnf..

Wed Apr 20 05:13:26 2016 - [info] MHA::MasterMonitor version 0.56.

Wed Apr 20 05:13:26 2016 - [info] GTID failover mode = 0

Wed Apr 20 05:13:26 2016 - [info] Dead Servers:

Wed Apr 20 05:13:26 2016 - [info] Alive Servers:

Wed Apr 20 05:13:26 2016 - [info] 192.168.10.130(192.168.10.130:3306)

Wed Apr 20 05:13:26 2016 - [info] 192.168.10.131(192.168.10.131:3306)

Wed Apr 20 05:13:26 2016 - [info] 192.168.10.132(192.168.10.132:3306)

Wed Apr 20 05:13:26 2016 - [info] 192.168.10.133(192.168.10.133:3306)

Wed Apr 20 05:13:26 2016 - [info] Alive Slaves:

Wed Apr 20 05:13:26 2016 - [info] 192.168.10.131(192.168.10.131:3306) Version=5.5.32-log (oldest major version between slaves) log-bin:enabled

Wed Apr 20 05:13:26 2016 - [info] Replicating from 192.168.10.130(192.168.10.130:3306)

Wed Apr 20 05:13:26 2016 - [info] Primary candidate for the new Master (candidate\_master is set)

Wed Apr 20 05:13:26 2016 - [info] 192.168.10.132(192.168.10.132:3306) Version=5.5.32-log (oldest major version between slaves) log-bin:enabled

Wed Apr 20 05:13:26 2016 - [info] Replicating from 192.168.10.130(192.168.10.130:3306)

Wed Apr 20 05:13:26 2016 - [info] 192.168.10.133(192.168.10.133:3306) Version=5.5.32-log (oldest major version between slaves) log-bin:enabled

Wed Apr 20 05:13:26 2016 - [info] Replicating from 192.168.10.130(192.168.10.130:3306)

Wed Apr 20 05:13:26 2016 - [info] Not candidate for the new Master (no\_master is set)

Wed Apr 20 05:13:26 2016 - [info] Current Alive Master: 192.168.10.130(192.168.10.130:3306)

Wed Apr 20 05:13:26 2016 - [info] Checking slave configurations..

Wed Apr 20 05:13:26 2016 - [info] read\_only=1 is not set on slave 192.168.10.131(192.168.10.131:3306).

Wed Apr 20 05:13:26 2016 - [info] Checking replication filtering settings..

Wed Apr 20 05:13:26 2016 - [info] binlog\_do\_db= , binlog\_ignore\_db=

Wed Apr 20 05:13:26 2016 - [info] Replication filtering check ok.

Wed Apr 20 05:13:26 2016 - [info] GTID (with auto-pos) is not supported

Wed Apr 20 05:13:26 2016 - [info] Starting SSH connection tests..

Wed Apr 20 05:13:28 2016 - [info] All SSH connection tests passed successfully.

Wed Apr 20 05:13:28 2016 - [info] Checking MHA Node version..

Wed Apr 20 05:13:28 2016 - [info] Version check ok.

Wed Apr 20 05:13:28 2016 - [info] Checking SSH publickey authentication settings on the current master..

Wed Apr 20 05:13:29 2016 - [info] HealthCheck: SSH to 192.168.10.130 is reachable.

Wed Apr 20 05:13:29 2016 - [info] Master MHA Node version is 0.56.

Wed Apr 20 05:13:29 2016 - [info] Checking recovery script configurations on 192.168.10.130(192.168.10.130:3306)..

Wed Apr 20 05:13:29 2016 - [info] Executing command: save\_binary\_logs --command=test --start\_pos=4 --binlog\_dir=/meishi/data/meishi --output\_file=/var/tmp/save\_binary\_logs\_test --manager\_version=0.56 --start\_file=mysql-bin.000006

Wed Apr 20 05:13:29 2016 - [info] Connecting to root@192.168.10.130(192.168.10.130:22)..

Creating /var/tmp if not exists.. ok.

Checking output directory is accessible or not..

ok.

Binlog found at /meishi/data/meishi, up to mysql-bin.000006

Wed Apr 20 05:13:29 2016 - [info] Binlog setting check done.

Wed Apr 20 05:13:29 2016 - [info] Checking SSH publickey authentication and checking recovery script configurations on all alive slave servers..

Wed Apr 20 05:13:29 2016 - [info] Executing command : apply\_diff\_relay\_logs --command=test --slave\_user='root' --slave\_host=192.168.10.131 --slave\_ip=192.168.10.131 --slave\_port=3306 --workdir=/var/tmp --target\_version=5.5.32-log --manager\_version=0.56 --relay\_log\_info=/meishi/data/meishi/relay-log.info --relay\_dir=/meishi/data/meishi/ --slave\_pass=xxx

Wed Apr 20 05:13:29 2016 - [info] Connecting to root@192.168.10.131(192.168.10.131:22)..

Checking slave recovery environment settings..

Opening /meishi/data/meishi/relay-log.info ... ok.

Relay log found at /meishi/data/meishi, up to relay-bin.000002

Temporary relay log file is /meishi/data/meishi/relay-bin.000002

Testing mysql connection and privileges.. done.

Testing mysqlbinlog output.. done.

Cleaning up test file(s).. done.

Wed Apr 20 05:13:29 2016 - [info] Executing command : apply\_diff\_relay\_logs --command=test --slave\_user='root' --slave\_host=192.168.10.132 --slave\_ip=192.168.10.132 --slave\_port=3306 --workdir=/var/tmp --target\_version=5.5.32-log --manager\_version=0.56 --relay\_log\_info=/meishi/data/meishi/relay-log.info --relay\_dir=/meishi/data/meishi/ --slave\_pass=xxx

Wed Apr 20 05:13:29 2016 - [info] Connecting to root@192.168.10.132(192.168.10.132:22)..

Checking slave recovery environment settings..

Opening /meishi/data/meishi/relay-log.info ... ok.

Relay log found at /meishi/data/meishi, up to relay-bin.000002

Temporary relay log file is /meishi/data/meishi/relay-bin.000002

Testing mysql connection and privileges.. done.

Testing mysqlbinlog output.. done.

Cleaning up test file(s).. done.

Wed Apr 20 05:13:29 2016 - [info] Executing command : apply\_diff\_relay\_logs --command=test --slave\_user='root' --slave\_host=192.168.10.133 --slave\_ip=192.168.10.133 --slave\_port=3306 --workdir=/var/tmp --target\_version=5.5.32-log --manager\_version=0.56 --relay\_log\_info=/meishi/data/meishi/relay-log.info --relay\_dir=/meishi/data/meishi/ --slave\_pass=xxx

Wed Apr 20 05:13:29 2016 - [info] Connecting to root@192.168.10.133(192.168.10.133:22)..

Checking slave recovery environment settings..

Opening /meishi/data/meishi/relay-log.info ... ok.

Relay log found at /meishi/data/meishi, up to relay-bin.000002

Temporary relay log file is /meishi/data/meishi/relay-bin.000002

Testing mysql connection and privileges.. done.

Testing mysqlbinlog output.. done.

Cleaning up test file(s).. done.

Wed Apr 20 05:13:29 2016 - [info] Slaves settings check done.

Wed Apr 20 05:13:29 2016 - [info]

192.168.10.130(192.168.10.130:3306) (current master)

+--192.168.10.131(192.168.10.131:3306)

+--192.168.10.132(192.168.10.132:3306)

+--192.168.10.133(192.168.10.133:3306)

Wed Apr 20 05:13:29 2016 - [info] Checking replication health on 192.168.10.131..

Wed Apr 20 05:13:29 2016 - [info] ok.

Wed Apr 20 05:13:29 2016 - [info] Checking replication health on 192.168.10.132..

Wed Apr 20 05:13:29 2016 - [info] ok.

Wed Apr 20 05:13:29 2016 - [info] Checking replication health on 192.168.10.133..

Wed Apr 20 05:13:29 2016 - [info] ok.

Wed Apr 20 05:13:29 2016 - [info] Checking master\_ip\_failover\_script status:

Wed Apr 20 05:13:29 2016 - [info] /etc/mha/scripts/master\_ip\_failover --command=status --ssh\_user=root --orig\_master\_host=192.168.10.130 --orig\_master\_ip=192.168.10.130 --orig\_master\_port=3306

Wed Apr 20 05:13:29 2016 - [info] OK.

Wed Apr 20 05:13:29 2016 - [warning] shutdown\_script is not defined.

Wed Apr 20 05:13:29 2016 - [info] Got exit code 0 (Not master dead).

MySQL Replication Health is OK.

5.5.3. 定时清理relay\_log日志

[root@manager ~]# masterha\_check\_status –conf=/etc/masterha/app1.cnf

app1 (pid:16179) is running(0:PING\_OK), master:192.168.1.101

[root@manager ~]# crontab -l

00 00 \* \* \* /usr/local/bin/purge\_relay\_logs –user=root –password=meishi disable\_relay\_log\_purge $amp;>amp;$gt; /masterha/purge\_relay\_logs.log 2$amp;>amp;$amp;1

[root@slave02 ~]# crontab -l

00 01 \* \* \* /usr/local/bin/purge\_relay\_logs –user=root –password=meishi –disable\_relay\_log\_purge $amp;>amp;$gt; /masterha/purge\_relay\_logs.log 2$amp;>amp;$amp;1

5.6. 启动MHA

[root@manager ~]# nohup /etc/mha/scripts/masterha\_manager --conf=/etc/mha/app1/app1.cnf &

[1] 48781

六. 测试流程操作

6.1 清空manager日志

[root@manager ~]# tailf /var/log/mha/app1/manager.log

[root@manager ~]# tailf /var/log/mha/app1/manager.log

6.2 启动MHA

6.2.1 nohup启动MHA

[root@manager ~]# nohup /etc/mha/scripts/masterha\_manager --conf=/etc/mha/app1/app1.cnf &

[1] 50177

[root@manager ~]#

6.2.2 查看启动日志

[root@manager ~]# tailf /var/log/mha/app1/manager.log

Wed Apr 20 05:35:56 2016 - [info] MHA::MasterMonitor version 0.56.

Wed Apr 20 05:35:56 2016 - [warning] /var/log/mha/app1/app1.master\_status.health already exists. You might have killed manager with SIGKILL(-9), may run two or more monitoring process for the same application, or use the same working directory. Check for details, and consider setting --workdir separately.

Wed Apr 20 05:35:56 2016 - [info] GTID failover mode = 0

Wed Apr 20 05:35:56 2016 - [info] Dead Servers:

Wed Apr 20 05:35:56 2016 - [info] Alive Servers:

Wed Apr 20 05:35:56 2016 - [info] 192.168.10.130(192.168.10.130:3306)

Wed Apr 20 05:35:56 2016 - [info] 192.168.10.131(192.168.10.131:3306)

Wed Apr 20 05:35:56 2016 - [info] 192.168.10.132(192.168.10.132:3306)

Wed Apr 20 05:35:56 2016 - [info] 192.168.10.133(192.168.10.133:3306)

Wed Apr 20 05:35:56 2016 - [info] Alive Slaves:

Wed Apr 20 05:35:56 2016 - [info] 192.168.10.131(192.168.10.131:3306) Version=5.5.32-log (oldest major version between slaves) log-bin:enabled

Wed Apr 20 05:35:56 2016 - [info] Replicating from 192.168.10.130(192.168.10.130:3306)

Wed Apr 20 05:35:56 2016 - [info] Primary candidate for the new Master (candidate\_master is set)

Wed Apr 20 05:35:56 2016 - [info] 192.168.10.132(192.168.10.132:3306) Version=5.5.32-log (oldest major version between slaves) log-bin:enabled

Wed Apr 20 05:35:56 2016 - [info] Replicating from 192.168.10.130(192.168.10.130:3306)

Wed Apr 20 05:35:56 2016 - [info] 192.168.10.133(192.168.10.133:3306) Version=5.5.32-log (oldest major version between slaves) log-bin:enabled

Wed Apr 20 05:35:56 2016 - [info] Replicating from 192.168.10.130(192.168.10.130:3306)

Wed Apr 20 05:35:56 2016 - [info] Not candidate for the new Master (no\_master is set)

Wed Apr 20 05:35:56 2016 - [info] Current Alive Master: 192.168.10.130(192.168.10.130:3306)

Wed Apr 20 05:35:56 2016 - [info] Checking slave configurations..

Wed Apr 20 05:35:56 2016 - [info] read\_only=1 is not set on slave 192.168.10.131(192.168.10.131:3306).

Wed Apr 20 05:35:56 2016 - [info] Checking replication filtering settings..

Wed Apr 20 05:35:56 2016 - [info] binlog\_do\_db= , binlog\_ignore\_db=

Wed Apr 20 05:35:56 2016 - [info] Replication filtering check ok.

Wed Apr 20 05:35:56 2016 - [info] GTID (with auto-pos) is not supported

Wed Apr 20 05:35:56 2016 - [info] Starting SSH connection tests..

Wed Apr 20 05:35:58 2016 - [info] All SSH connection tests passed successfully.

Wed Apr 20 05:35:58 2016 - [info] Checking MHA Node version..

Wed Apr 20 05:35:58 2016 - [info] Version check ok.

Wed Apr 20 05:35:58 2016 - [info] Checking SSH publickey authentication settings on the current master..

Wed Apr 20 05:35:58 2016 - [info] HealthCheck: SSH to 192.168.10.130 is reachable.

Wed Apr 20 05:35:59 2016 - [info] Master MHA Node version is 0.56.

Wed Apr 20 05:35:59 2016 - [info] Checking recovery script configurations on 192.168.10.130(192.168.10.130:3306)..

Wed Apr 20 05:35:59 2016 - [info] Executing command: save\_binary\_logs --command=test --start\_pos=4 --binlog\_dir=/meishi/data/meishi --output\_file=/var/tmp/save\_binary\_logs\_test --manager\_version=0.56 --start\_file=mysql-bin.000006

Wed Apr 20 05:35:59 2016 - [info] Connecting to root@192.168.10.130(192.168.10.130:22)..

Creating /var/tmp if not exists.. ok.

Checking output directory is accessible or not..

ok.

Binlog found at /meishi/data/meishi, up to mysql-bin.000006

Wed Apr 20 05:35:59 2016 - [info] Binlog setting check done.

Wed Apr 20 05:35:59 2016 - [info] Checking SSH publickey authentication and checking recovery script configurations on all alive slave servers..

Wed Apr 20 05:35:59 2016 - [info] Executing command : apply\_diff\_relay\_logs --command=test --slave\_user='root' --slave\_host=192.168.10.131 --slave\_ip=192.168.10.131 --slave\_port=3306 --workdir=/var/tmp --target\_version=5.5.32-log --manager\_version=0.56 --relay\_log\_info=/meishi/data/meishi/relay-log.info --relay\_dir=/meishi/data/meishi/ --slave\_pass=xxx

Wed Apr 20 05:35:59 2016 - [info] Connecting to root@192.168.10.131(192.168.10.131:22)..

Checking slave recovery environment settings..

Opening /meishi/data/meishi/relay-log.info ... ok.

Relay log found at /meishi/data/meishi, up to relay-bin.000002

Temporary relay log file is /meishi/data/meishi/relay-bin.000002

Testing mysql connection and privileges.. done.

Testing mysqlbinlog output.. done.

Cleaning up test file(s).. done.

Wed Apr 20 05:35:59 2016 - [info] Executing command : apply\_diff\_relay\_logs --command=test --slave\_user='root' --slave\_host=192.168.10.132 --slave\_ip=192.168.10.132 --slave\_port=3306 --workdir=/var/tmp --target\_version=5.5.32-log --manager\_version=0.56 --relay\_log\_info=/meishi/data/meishi/relay-log.info --relay\_dir=/meishi/data/meishi/ --slave\_pass=xxx

Wed Apr 20 05:35:59 2016 - [info] Connecting to root@192.168.10.132(192.168.10.132:22)..

Checking slave recovery environment settings..

Opening /meishi/data/meishi/relay-log.info ... ok.

Relay log found at /meishi/data/meishi, up to relay-bin.000002

Temporary relay log file is /meishi/data/meishi/relay-bin.000002

Testing mysql connection and privileges.. done.

Testing mysqlbinlog output.. done.

Cleaning up test file(s).. done.

Wed Apr 20 05:35:59 2016 - [info] Executing command : apply\_diff\_relay\_logs --command=test --slave\_user='root' --slave\_host=192.168.10.133 --slave\_ip=192.168.10.133 --slave\_port=3306 --workdir=/var/tmp --target\_version=5.5.32-log --manager\_version=0.56 --relay\_log\_info=/meishi/data/meishi/relay-log.info --relay\_dir=/meishi/data/meishi/ --slave\_pass=xxx

Wed Apr 20 05:35:59 2016 - [info] Connecting to root@192.168.10.133(192.168.10.133:22)..

Checking slave recovery environment settings..

Opening /meishi/data/meishi/relay-log.info ... ok.

Relay log found at /meishi/data/meishi, up to relay-bin.000002

Temporary relay log file is /meishi/data/meishi/relay-bin.000002

Testing mysql connection and privileges.. done.

Testing mysqlbinlog output.. done.

Cleaning up test file(s).. done.

Wed Apr 20 05:35:59 2016 - [info] Slaves settings check done.

Wed Apr 20 05:35:59 2016 - [info]

192.168.10.130(192.168.10.130:3306) (current master)

+--192.168.10.131(192.168.10.131:3306)

+--192.168.10.132(192.168.10.132:3306)

+--192.168.10.133(192.168.10.133:3306)

Wed Apr 20 05:35:59 2016 - [info] Checking master\_ip\_failover\_script status:

Wed Apr 20 05:35:59 2016 - [info] /etc/mha/scripts/master\_ip\_failover --command=status --ssh\_user=root --orig\_master\_host=192.168.10.130 --orig\_master\_ip=192.168.10.130 --orig\_master\_port=3306

Wed Apr 20 05:35:59 2016 - [info] OK.

Wed Apr 20 05:35:59 2016 - [warning] shutdown\_script is not defined.

Wed Apr 20 05:35:59 2016 - [info] Set master ping interval 1 seconds.

Wed Apr 20 05:35:59 2016 - [warning] secondary\_check\_script is not defined. It is highly recommended setting it to check master reachability from two or more routes.

Wed Apr 20 05:35:59 2016 - [info] Starting ping health check on 192.168.10.130(192.168.10.130:3306)..

Wed Apr 20 05:35:59 2016 - [info] Ping(SELECT) succeeded, waiting until MySQL doesn't respond..

6.3 关闭主库的MySQL

6.3.1 杀死mysql进程

[root@master-server ~]# pkill mysql

[root@master-server ~]# netstat -lnpt |grep mysql

6.3.2 查看manager日志

Wed Apr 20 05:40:15 2016 - [info] Fetching dead master's binary logs..

Wed Apr 20 05:40:15 2016 - [info] Executing command on the dead master 192.168.10.130(192.168.10.130:3306): save\_binary\_logs --command=save --start\_file=mysql-bin.000006 --start\_pos=107 --binlog\_dir=/meishi/data/meishi --output\_file=/var/tmp/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog --handle\_raw\_binlog=1 --disable\_log\_bin=0 --manager\_version=0.56

Creating /var/tmp if not exists.. ok.

Concat binary/relay logs from mysql-bin.000006 pos 107 to mysql-bin.000006 EOF into /var/tmp/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog ..

Dumping binlog format description event, from position 0 to 107.. ok.

Dumping effective binlog data from /meishi/data/meishi/mysql-bin.000006 position 107 to tail(126).. ok.

Concat succeeded.

Wed Apr 20 05:40:15 2016 - [info] scp from root@192.168.10.130:/var/tmp/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog to local:/var/log/mha/app1/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog succeeded.

Wed Apr 20 05:40:15 2016 - [info] HealthCheck: SSH to 192.168.10.131 is reachable.

Wed Apr 20 05:40:16 2016 - [info] HealthCheck: SSH to 192.168.10.132 is reachable.

Wed Apr 20 05:40:16 2016 - [info] HealthCheck: SSH to 192.168.10.133 is reachable.

Wed Apr 20 05:40:16 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] \* Phase 3.3: Determining New Master Phase..

Wed Apr 20 05:40:16 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] Finding the latest slave that has all relay logs for recovering other slaves..

Wed Apr 20 05:40:16 2016 - [info] All slaves received relay logs to the same position. No need to resync each other.

Wed Apr 20 05:40:16 2016 - [info] Searching new master from slaves..

Wed Apr 20 05:40:16 2016 - [info] Candidate masters from the configuration file:

Wed Apr 20 05:40:16 2016 - [info] 192.168.10.131(192.168.10.131:3306) Version=5.5.32-log (oldest major version between slaves) log-bin:enabled

Wed Apr 20 05:40:16 2016 - [info] Replicating from 192.168.10.130(192.168.10.130:3306)

Wed Apr 20 05:40:16 2016 - [info] Primary candidate for the new Master (candidate\_master is set)

Wed Apr 20 05:40:16 2016 - [info] Non-candidate masters:

Wed Apr 20 05:40:16 2016 - [info] 192.168.10.133(192.168.10.133:3306) Version=5.5.32-log (oldest major version between slaves) log-bin:enabled

Wed Apr 20 05:40:16 2016 - [info] Replicating from 192.168.10.130(192.168.10.130:3306)

Wed Apr 20 05:40:16 2016 - [info] Not candidate for the new Master (no\_master is set)

Wed Apr 20 05:40:16 2016 - [info] Searching from candidate\_master slaves which have received the latest relay log events..

Wed Apr 20 05:40:16 2016 - [info] New master is 192.168.10.131(192.168.10.131:3306)

Wed Apr 20 05:40:16 2016 - [info] Starting master failover..

Wed Apr 20 05:40:16 2016 - [info]

From:

192.168.10.130(192.168.10.130:3306) (current master)

+--192.168.10.131(192.168.10.131:3306)

+--192.168.10.132(192.168.10.132:3306)

+--192.168.10.133(192.168.10.133:3306)

To:

192.168.10.131(192.168.10.131:3306) (new master)

+--192.168.10.132(192.168.10.132:3306)

+--192.168.10.133(192.168.10.133:3306)

Wed Apr 20 05:40:16 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] \* Phase 3.3: New Master Diff Log Generation Phase..

Wed Apr 20 05:40:16 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] This server has all relay logs. No need to generate diff files from the latest slave.

Wed Apr 20 05:40:16 2016 - [info] Sending binlog..

Wed Apr 20 05:40:16 2016 - [info] scp from local:/var/log/mha/app1/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog to root@192.168.10.131:/var/tmp/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog succeeded.

Wed Apr 20 05:40:16 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] \* Phase 3.4: Master Log Apply Phase..

Wed Apr 20 05:40:16 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] \*NOTICE: If any error happens from this phase, manual recovery is needed.

Wed Apr 20 05:40:16 2016 - [info] Starting recovery on 192.168.10.131(192.168.10.131:3306)..

Wed Apr 20 05:40:16 2016 - [info] Generating diffs succeeded.

Wed Apr 20 05:40:16 2016 - [info] Waiting until all relay logs are applied.

Wed Apr 20 05:40:16 2016 - [info] done.

Wed Apr 20 05:40:16 2016 - [info] Getting slave status..

Wed Apr 20 05:40:16 2016 - [info] This slave(192.168.10.131)'s Exec\_Master\_Log\_Pos equals to Read\_Master\_Log\_Pos(mysql-bin.000006:107). No need to recover from Exec\_Master\_Log\_Pos.

Wed Apr 20 05:40:16 2016 - [info] Connecting to the target slave host 192.168.10.131, running recover script..

Wed Apr 20 05:40:16 2016 - [info] Executing command: apply\_diff\_relay\_logs --command=apply --slave\_user='root' --slave\_host=192.168.10.131 --slave\_ip=192.168.10.131 --slave\_port=3306 --apply\_files=/var/tmp/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog --workdir=/var/tmp --target\_version=5.5.32-log --timestamp=20160420054014 --handle\_raw\_binlog=1 --disable\_log\_bin=0 --manager\_version=0.56 --slave\_pass=xxx

Wed Apr 20 05:40:16 2016 - [info]

Applying differential binary/relay log files /var/tmp/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog on 192.168.10.131:3306. This may take long time...

Applying log files succeeded.

Wed Apr 20 05:40:16 2016 - [info] All relay logs were successfully applied.

Wed Apr 20 05:40:16 2016 - [info] Getting new master's binlog name and position..

Wed Apr 20 05:40:16 2016 - [info] mysql-bin.000003:107

Wed Apr 20 05:40:16 2016 - [info] All other slaves should start replication from here. Statement should be: CHANGE MASTER TO MASTER\_HOST='**192.168.10.131**', MASTER\_PORT=3306, MASTER\_LOG\_FILE='**mysql-bin.000003**', **MASTER\_LOG\_POS=107**, MASTER\_USER='rep', MASTER\_PASSWORD='xxx';

Wed Apr 20 05:40:16 2016 - [info] Executing master IP activate script:

Wed Apr 20 05:40:16 2016 - [info] /etc/mha/scripts/master\_ip\_failover --command=start --ssh\_user=root --orig\_master\_host=192.168.10.130 --orig\_master\_ip=192.168.10.130 --orig\_master\_port=3306 --new\_master\_host=192.168.10.131 --new\_master\_ip=192.168.10.131 --new\_master\_port=3306 --new\_master\_user='root' --new\_master\_password='meishi'

Undefined subroutine &main::FIXME\_xxx\_create\_user called at /etc/mha/scripts/master\_ip\_failover line 88.

Set read\_only=0 on the new master.

Creating app user on the new master..

Wed Apr 20 05:40:16 2016 - [error][/usr/share/perl5/vendor\_perl/MHA/MasterFailover.pm, ln1588] Failed to activate master IP address for 192.168.10.131(192.168.10.131:3306) with return code 10:0

Wed Apr 20 05:40:16 2016 - [warning] Proceeding.

Wed Apr 20 05:40:16 2016 - [info] \*\* Finished master recovery successfully.

Wed Apr 20 05:40:16 2016 - [info] \* Phase 3: Master Recovery Phase completed.

Wed Apr 20 05:40:16 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] \* Phase 4: Slaves Recovery Phase..

Wed Apr 20 05:40:16 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] \* Phase 4.1: Starting Parallel Slave Diff Log Generation Phase..

Wed Apr 20 05:40:16 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] -- Slave diff file generation on host 192.168.10.132(192.168.10.132:3306) started, pid: 50737. Check tmp log /var/log/mha/app1/192.168.10.132\_3306\_20160420054014.log if it takes time..

Wed Apr 20 05:40:16 2016 - [info] -- Slave diff file generation on host 192.168.10.133(192.168.10.133:3306) started, pid: 50738. Check tmp log /var/log/mha/app1/192.168.10.133\_3306\_20160420054014.log if it takes time..

Wed Apr 20 05:40:16 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] Log messages from 192.168.10.133 ...

Wed Apr 20 05:40:16 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] This server has all relay logs. No need to generate diff files from the latest slave.

Wed Apr 20 05:40:16 2016 - [info] End of log messages from 192.168.10.133.

Wed Apr 20 05:40:16 2016 - [info] -- 192.168.10.133(192.168.10.133:3306) has the latest relay log events.

Wed Apr 20 05:40:16 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] Log messages from 192.168.10.132 ...

Wed Apr 20 05:40:16 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] This server has all relay logs. No need to generate diff files from the latest slave.

Wed Apr 20 05:40:16 2016 - [info] End of log messages from 192.168.10.132.

Wed Apr 20 05:40:16 2016 - [info] -- 192.168.10.132(192.168.10.132:3306) has the latest relay log events.

Wed Apr 20 05:40:16 2016 - [info] Generating relay diff files from the latest slave succeeded.

Wed Apr 20 05:40:16 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] \* Phase 4.2: Starting Parallel Slave Log Apply Phase..

Wed Apr 20 05:40:16 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] -- Slave recovery on host 192.168.10.132(192.168.10.132:3306) started, pid: 50741. Check tmp log /var/log/mha/app1/192.168.10.132\_3306\_20160420054014.log if it takes time..

Wed Apr 20 05:40:16 2016 - [info] -- Slave recovery on host 192.168.10.133(192.168.10.133:3306) started, pid: 50742. Check tmp log /var/log/mha/app1/192.168.10.133\_3306\_20160420054014.log if it takes time..

Wed Apr 20 05:40:17 2016 - [info]

Wed Apr 20 05:40:17 2016 - [info] Log messages from 192.168.10.133 ...

Wed Apr 20 05:40:17 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] Sending binlog..

Wed Apr 20 05:40:16 2016 - [info] scp from local:/var/log/mha/app1/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog to root@192.168.10.133:/var/tmp/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog succeeded.

Wed Apr 20 05:40:16 2016 - [info] Starting recovery on 192.168.10.133(192.168.10.133:3306)..

Wed Apr 20 05:40:16 2016 - [info] Generating diffs succeeded.

Wed Apr 20 05:40:16 2016 - [info] Waiting until all relay logs are applied.

Wed Apr 20 05:40:16 2016 - [info] done.

Wed Apr 20 05:40:16 2016 - [info] Getting slave status..

Wed Apr 20 05:40:16 2016 - [info] This slave(192.168.10.133)'s Exec\_Master\_Log\_Pos equals to Read\_Master\_Log\_Pos(mysql-bin.000006:107). No need to recover from Exec\_Master\_Log\_Pos.

Wed Apr 20 05:40:16 2016 - [info] Connecting to the target slave host 192.168.10.133, running recover script..

Wed Apr 20 05:40:16 2016 - [info] Executing command: apply\_diff\_relay\_logs --command=apply --slave\_user='root' --slave\_host=192.168.10.133 --slave\_ip=192.168.10.133 --slave\_port=3306 --apply\_files=/var/tmp/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog --workdir=/var/tmp --target\_version=5.5.32-log --timestamp=20160420054014 --handle\_raw\_binlog=1 --disable\_log\_bin=0 --manager\_version=0.56 --slave\_pass=xxx

Wed Apr 20 05:40:17 2016 - [info]

Applying differential binary/relay log files /var/tmp/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog on 192.168.10.133:3306. This may take long time...

Applying log files succeeded.

Wed Apr 20 05:40:17 2016 - [info] All relay logs were successfully applied.

Wed Apr 20 05:40:17 2016 - [info] Resetting slave 192.168.10.133(192.168.10.133:3306) and starting replication from the new master 192.168.10.131(192.168.10.131:3306)..

Wed Apr 20 05:40:17 2016 - [info] Executed CHANGE MASTER.

Wed Apr 20 05:40:17 2016 - [info] Slave started.

Wed Apr 20 05:40:17 2016 - [info] End of log messages from 192.168.10.133.

Wed Apr 20 05:40:17 2016 - [info] -- Slave recovery on host 192.168.10.133(192.168.10.133:3306) succeeded.

Wed Apr 20 05:40:17 2016 - [info]

Wed Apr 20 05:40:17 2016 - [info] Log messages from 192.168.10.132 ...

Wed Apr 20 05:40:17 2016 - [info]

Wed Apr 20 05:40:16 2016 - [info] Sending binlog..

Wed Apr 20 05:40:17 2016 - [info] scp from local:/var/log/mha/app1/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog to root@192.168.10.132:/var/tmp/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog succeeded.

Wed Apr 20 05:40:17 2016 - [info] Starting recovery on 192.168.10.132(192.168.10.132:3306)..

Wed Apr 20 05:40:17 2016 - [info] Generating diffs succeeded.

Wed Apr 20 05:40:17 2016 - [info] Waiting until all relay logs are applied.

Wed Apr 20 05:40:17 2016 - [info] done.

Wed Apr 20 05:40:17 2016 - [info] Getting slave status..

Wed Apr 20 05:40:17 2016 - [info] This slave(192.168.10.132)'s Exec\_Master\_Log\_Pos equals to Read\_Master\_Log\_Pos(mysql-bin.000006:107). No need to recover from Exec\_Master\_Log\_Pos.

Wed Apr 20 05:40:17 2016 - [info] Connecting to the target slave host 192.168.10.132, running recover script..

Wed Apr 20 05:40:17 2016 - [info] Executing command: apply\_diff\_relay\_logs --command=apply --slave\_user='root' --slave\_host=192.168.10.132 --slave\_ip=192.168.10.132 --slave\_port=3306 --apply\_files=/var/tmp/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog --workdir=/var/tmp --target\_version=5.5.32-log --timestamp=20160420054014 --handle\_raw\_binlog=1 --disable\_log\_bin=0 --manager\_version=0.56 --slave\_pass=xxx

Wed Apr 20 05:40:17 2016 - [info]

Applying differential binary/relay log files /var/tmp/saved\_master\_binlog\_from\_192.168.10.130\_3306\_20160420054014.binlog on 192.168.10.132:3306. This may take long time...

Applying log files succeeded.

Wed Apr 20 05:40:17 2016 - [info] All relay logs were successfully applied.

Wed Apr 20 05:40:17 2016 - [info] Resetting slave 192.168.10.132(192.168.10.132:3306) and starting replication from the new master 192.168.10.131(192.168.10.131:3306)..

Wed Apr 20 05:40:17 2016 - [info] Executed CHANGE MASTER.

Wed Apr 20 05:40:17 2016 - [info] Slave started.

Wed Apr 20 05:40:17 2016 - [info] End of log messages from 192.168.10.132.

Wed Apr 20 05:40:17 2016 - [info] -- Slave recovery on host 192.168.10.132(192.168.10.132:3306) succeeded.

Wed Apr 20 05:40:17 2016 - [info] All new slave servers recovered successfully.

Wed Apr 20 05:40:17 2016 - [info]

Wed Apr 20 05:40:17 2016 - [info] \* Phase 5: New master cleanup phase..

Wed Apr 20 05:40:17 2016 - [info]

Wed Apr 20 05:40:17 2016 - [info] Resetting slave info on the new master..

Wed Apr 20 05:40:17 2016 - [info] 192.168.10.131: Resetting slave info succeeded.

Wed Apr 20 05:40:17 2016 - [info] Master failover to 192.168.10.131(192.168.10.131:3306) completed successfully.

Wed Apr 20 05:40:17 2016 - [info]

----- Failover Report -----

app1: MySQL Master failover 192.168.10.130(192.168.10.130:3306) to 192.168.10.131(192.168.10.131:3306) succeeded

Master 192.168.10.130(192.168.10.130:3306) is down!

Check MHA Manager logs at manager:/var/log/mha/app1/manager.log for details.

Started automated(non-interactive) failover.

Invalidated master IP address on 192.168.10.130(192.168.10.130:3306)

The latest slave 192.168.10.131(192.168.10.131:3306) has all relay logs for recovery.

Selected 192.168.10.131(192.168.10.131:3306) as a new master.

192.168.10.131(192.168.10.131:3306): OK: Applying all logs succeeded.

Failed to activate master IP address for 192.168.10.131(192.168.10.131:3306) with return code 10:0

192.168.10.133(192.168.10.133:3306): This host has the latest relay log events.

192.168.10.132(192.168.10.132:3306): This host has the latest relay log events.

Generating relay diff files from the latest slave succeeded.

192.168.10.133(192.168.10.133:3306): OK: Applying all logs succeeded. Slave started, replicating from 192.168.10.131(192.168.10.131:3306)

192.168.10.132(192.168.10.132:3306): OK: Applying all logs succeeded. Slave started, replicating from 192.168.10.131(192.168.10.131:3306)

192.168.10.131(192.168.10.131:3306): Resetting slave info succeeded.

Master failover to 192.168.10.131(192.168.10.131:3306) completed successfully.

6.3.3 查看从库同步的主库

[root@slave02 ~]# mysql -uroot -pmeishi -h 192.168.10.132

mysql> show slave status\G

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. row \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Slave\_IO\_State: Waiting for master to send event

Master\_Host: **192.168.10.131**

Master\_User: rep

Master\_Port: 3306

Connect\_Retry: 60

Master\_Log\_File: mysql-bin.000003

Read\_Master\_Log\_Pos: 107

Relay\_Log\_File: relay-bin.000002

Relay\_Log\_Pos: 253

Relay\_Master\_Log\_File: mysql-bin.000003

Slave\_IO\_Running: Yes

Slave\_SQL\_Running: Yes

Replicate\_Do\_DB:

Replicate\_Ignore\_DB: mysql

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: 107

Relay\_Log\_Space: 403

Until\_Condition: None

Until\_Log\_File:

Until\_Log\_Pos: 0

Master\_SSL\_Allowed: No

Master\_SSL\_CA\_File:

Master\_SSL\_CA\_Path:

Master\_SSL\_Cert:

Master\_SSL\_Cipher:

Master\_SSL\_Key:

Seconds\_Behind\_Master: 0

Master\_SSL\_Verify\_Server\_Cert: No

Last\_IO\_Errno: 0

Last\_IO\_Error:

Last\_SQL\_Errno: 0

Last\_SQL\_Error:

Replicate\_Ignore\_Server\_Ids:

Master\_Server\_Id: 2

1 row in set (0.00 sec)

**可以看到主主库停掉，将主据库迁移到库192.168.10.131，其他从库也都同步到备主库192.168.10.131.**

七、在线恢复主主库

7.1 启动主主库MySQL

[root@master-server ~]# /etc/init.d/mysql start

Starting MySQL...

[root@master-server ~]# netstat -lnpt |grep mysql

tcp 0 0 0.0.0.0:3306 0.0.0.0:\* LISTEN 50400/mysqld

[root@master-server ~]#

7.2 将主主库同步到备主库

CHANGE MASTER TO

MASTER\_HOST='192.168.10.130',

MASTER\_PORT=3306,

MASTER\_USER='rep',

MASTER\_PASSWORD='rep',

MASTER\_LOG\_FILE='mysql-bin.000003',

MASTER\_LOG\_POS=107;

mysql> CHANGE MASTER TO MASTER\_HOST='192.168.10.131', MASTER\_PORT=3306, MASTER\_LOG\_FILE='mysql-bin.000003', MASTER\_LOG\_POS=107, MASTER\_USER='rep',MASTER\_PASSWORD='rep';

Query OK, 0 rows affected (0.01 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: **192.168.10.131**

Master\_User: rep

Master\_Port: 3306

Connect\_Retry: 60

Master\_Log\_File: mysql-bin.000003

Read\_Master\_Log\_Pos: 107

Relay\_Log\_File: relay-bin.000002

Relay\_Log\_Pos: 253

Relay\_Master\_Log\_File: mysql-bin.000003

Slave\_IO\_Running: Yes

Slave\_SQL\_Running: Yes

Replicate\_Do\_DB:

Replicate\_Ignore\_DB: mysql

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: 107

Relay\_Log\_Space: 403

Until\_Condition: None

Until\_Log\_File:

Until\_Log\_Pos: 0

Master\_SSL\_Allowed: No

Master\_SSL\_CA\_File:

Master\_SSL\_CA\_Path:

Master\_SSL\_Cert:

Master\_SSL\_Cipher:

Master\_SSL\_Key:

Seconds\_Behind\_Master: 0

Master\_SSL\_Verify\_Server\_Cert: No

Last\_IO\_Errno: 0

Last\_IO\_Error:

Last\_SQL\_Errno: 0

Last\_SQL\_Error:

Replicate\_Ignore\_Server\_Ids:

Master\_Server\_Id: 2

1 row in set (0.00 sec)

7.3 恢复主主库

masterha\_master\_switch --conf=/etc/masterha/app1.cnf --master\_state=alive --new\_master\_host=192.168.10.130 --new\_master\_port=3306 --orig\_master\_is\_new\_slave --running\_updates\_limit=10000

[root@manager ~]# /etc/mha/scripts/masterha\_master\_switch --conf=/etc/mha/app1/app1.cnf --master\_state=alive --new\_master\_host=192.168.10.130 --new\_master\_port=3306 --orig\_master\_is\_new\_slave --running\_updates\_limit=10000

Wed Apr 20 06:04:01 2016 - [info] MHA::MasterRotate version 0.56.

Wed Apr 20 06:04:01 2016 - [info] Starting online master switch..

Wed Apr 20 06:04:01 2016 - [info]

Wed Apr 20 06:04:01 2016 - [info] \* Phase 1: Configuration Check Phase..

Wed Apr 20 06:04:01 2016 - [info]

Wed Apr 20 06:04:01 2016 - [warning] Global configuration file /etc/masterha\_default.cnf not found. Skipping.

Wed Apr 20 06:04:01 2016 - [info] Reading application default configuration from /etc/mha/app1/app1.cnf..

Wed Apr 20 06:04:01 2016 - [info] Reading server configuration from /etc/mha/app1/app1.cnf..

Wed Apr 20 06:04:01 2016 - [info] GTID failover mode = 0

Wed Apr 20 06:04:01 2016 - [info] Current Alive Master: 192.168.10.131(192.168.10.131:3306)

Wed Apr 20 06:04:01 2016 - [info] Alive Slaves:

Wed Apr 20 06:04:01 2016 - [info] 192.168.10.130(192.168.10.130:3306) Version=5.5.32-log (oldest major version between slaves) log-bin:enabled

Wed Apr 20 06:04:01 2016 - [info] Replicating from 192.168.10.131(192.168.10.131:3306)

Wed Apr 20 06:04:01 2016 - [info] Primary candidate for the new Master (candidate\_master is set)

Wed Apr 20 06:04:01 2016 - [info] 192.168.10.132(192.168.10.132:3306) Version=5.5.32-log (oldest major version between slaves) log-bin:enabled

Wed Apr 20 06:04:01 2016 - [info] Replicating from 192.168.10.131(192.168.10.131:3306)

Wed Apr 20 06:04:01 2016 - [info] 192.168.10.133(192.168.10.133:3306) Version=5.5.32-log (oldest major version between slaves) log-bin:enabled

Wed Apr 20 06:04:01 2016 - [info] Replicating from 192.168.10.131(192.168.10.131:3306)

Wed Apr 20 06:04:01 2016 - [info] Not candidate for the new Master (no\_master is set)

It is better to execute FLUSH NO\_WRITE\_TO\_BINLOG TABLES on the master before switching. Is it ok to execute on 192.168.10.131(192.168.10.131:3306)? (YES/no): yes

Wed Apr 20 06:04:02 2016 - [info] Executing FLUSH NO\_WRITE\_TO\_BINLOG TABLES. This may take long time..

Wed Apr 20 06:04:02 2016 - [info] ok.

Wed Apr 20 06:04:02 2016 - [info] Checking MHA is not monitoring or doing failover..

Wed Apr 20 06:04:02 2016 - [info] Checking replication health on 192.168.10.130..

Wed Apr 20 06:04:02 2016 - [info] ok.

Wed Apr 20 06:04:02 2016 - [info] Checking replication health on 192.168.10.132..

Wed Apr 20 06:04:02 2016 - [info] ok.

Wed Apr 20 06:04:02 2016 - [info] Checking replication health on 192.168.10.133..

Wed Apr 20 06:04:02 2016 - [info] ok.

Wed Apr 20 06:04:02 2016 - [info] 192.168.10.130 can be new master.

Wed Apr 20 06:04:02 2016 - [info]

From:

192.168.10.131(192.168.10.131:3306) (current master)

+--192.168.10.130(192.168.10.130:3306)

+--192.168.10.132(192.168.10.132:3306)

+--192.168.10.133(192.168.10.133:3306)

To:

192.168.10.130(192.168.10.130:3306) (new master)

+--192.168.10.132(192.168.10.132:3306)

+--192.168.10.133(192.168.10.133:3306)

+--192.168.10.131(192.168.10.131:3306)

Starting master switch from 192.168.10.131(192.168.10.131:3306) to 192.168.10.130(192.168.10.130:3306)? (yes/NO): yes

Wed Apr 20 06:04:05 2016 - [info] Checking whether 192.168.10.130(192.168.10.130:3306) is ok for the new master..

Wed Apr 20 06:04:05 2016 - [info] ok.

Wed Apr 20 06:04:05 2016 - [info] 192.168.10.131(192.168.10.131:3306): SHOW SLAVE STATUS returned empty result. To check replication filtering rules, temporarily executing CHANGE MASTER to a dummy host.

Wed Apr 20 06:04:05 2016 - [info] 192.168.10.131(192.168.10.131:3306): Resetting slave pointing to the dummy host.

Wed Apr 20 06:04:05 2016 - [info] \*\* Phase 1: Configuration Check Phase completed.

Wed Apr 20 06:04:05 2016 - [info]

Wed Apr 20 06:04:05 2016 - [info] \* Phase 2: Rejecting updates Phase..

Wed Apr 20 06:04:05 2016 - [info]

master\_ip\_online\_change\_script is not defined. If you do not disable writes on the current master manually, applications keep writing on the current master. Is it ok to proceed? (yes/NO): yes

Wed Apr 20 06:04:07 2016 - [info] Locking all tables on the orig master to reject updates from everybody (including root):

Wed Apr 20 06:04:07 2016 - [info] Executing FLUSH TABLES WITH READ LOCK..

Wed Apr 20 06:04:07 2016 - [info] ok.

Wed Apr 20 06:04:07 2016 - [info] Orig master binlog:pos is mysql-bin.000003:107.

Wed Apr 20 06:04:07 2016 - [info] Waiting to execute all relay logs on 192.168.10.130(192.168.10.130:3306)..

Wed Apr 20 06:04:07 2016 - [info] master\_pos\_wait(mysql-bin.000003:107) completed on 192.168.10.130(192.168.10.130:3306). Executed 0 events.

Wed Apr 20 06:04:07 2016 - [info] done.

Wed Apr 20 06:04:07 2016 - [info] Getting new master's binlog name and position..

Wed Apr 20 06:04:07 2016 - [info] mysql-bin.000007:107

Wed Apr 20 06:04:07 2016 - [info] All other slaves should start replication from here. Statement should be: CHANGE MASTER TO MASTER\_HOST='192.168.10.130', MASTER\_PORT=3306, MASTER\_LOG\_FILE='mysql-bin.000007', MASTER\_LOG\_POS=107, MASTER\_USER='rep', MASTER\_PASSWORD='xxx';

Wed Apr 20 06:04:07 2016 - [info]

Wed Apr 20 06:04:07 2016 - [info] \* Switching slaves in parallel..

Wed Apr 20 06:04:07 2016 - [info]

Wed Apr 20 06:04:07 2016 - [info] -- Slave switch on host 192.168.10.132(192.168.10.132:3306) started, pid: 50847

Wed Apr 20 06:04:07 2016 - [info]

Wed Apr 20 06:04:07 2016 - [info] -- Slave switch on host 192.168.10.133(192.168.10.133:3306) started, pid: 50848

Wed Apr 20 06:04:07 2016 - [info]

Wed Apr 20 06:04:07 2016 - [info] Log messages from 192.168.10.132 ...

Wed Apr 20 06:04:07 2016 - [info]

Wed Apr 20 06:04:07 2016 - [info] Waiting to execute all relay logs on 192.168.10.132(192.168.10.132:3306)..

Wed Apr 20 06:04:07 2016 - [info] master\_pos\_wait(mysql-bin.000003:107) completed on 192.168.10.132(192.168.10.132:3306). Executed 0 events.

Wed Apr 20 06:04:07 2016 - [info] done.

Wed Apr 20 06:04:07 2016 - [info] Resetting slave 192.168.10.132(192.168.10.132:3306) and starting replication from the new master 192.168.10.130(192.168.10.130:3306)..

Wed Apr 20 06:04:07 2016 - [info] Executed CHANGE MASTER.

Wed Apr 20 06:04:07 2016 - [info] Slave started.

Wed Apr 20 06:04:07 2016 - [info] End of log messages from 192.168.10.132 ...

Wed Apr 20 06:04:07 2016 - [info]

Wed Apr 20 06:04:07 2016 - [info] -- Slave switch on host 192.168.10.132(192.168.10.132:3306) succeeded.

Wed Apr 20 06:04:07 2016 - [info] Log messages from 192.168.10.133 ...

Wed Apr 20 06:04:07 2016 - [info]

Wed Apr 20 06:04:07 2016 - [info] Waiting to execute all relay logs on 192.168.10.133(192.168.10.133:3306)..

Wed Apr 20 06:04:07 2016 - [info] master\_pos\_wait(mysql-bin.000003:107) completed on 192.168.10.133(192.168.10.133:3306). Executed 0 events.

Wed Apr 20 06:04:07 2016 - [info] done.

Wed Apr 20 06:04:07 2016 - [info] Resetting slave 192.168.10.133(192.168.10.133:3306) and starting replication from the new master 192.168.10.130(192.168.10.130:3306)..

Wed Apr 20 06:04:07 2016 - [info] Executed CHANGE MASTER.

Wed Apr 20 06:04:07 2016 - [info] Slave started.

Wed Apr 20 06:04:07 2016 - [info] End of log messages from 192.168.10.133 ...

Wed Apr 20 06:04:07 2016 - [info]

Wed Apr 20 06:04:07 2016 - [info] -- Slave switch on host 192.168.10.133(192.168.10.133:3306) succeeded.

Wed Apr 20 06:04:07 2016 - [info] Unlocking all tables on the orig master:

Wed Apr 20 06:04:07 2016 - [info] Executing UNLOCK TABLES..

Wed Apr 20 06:04:07 2016 - [info] ok.

Wed Apr 20 06:04:07 2016 - [info] Starting orig master as a new slave..

Wed Apr 20 06:04:07 2016 - [info] Resetting slave 192.168.10.131(192.168.10.131:3306) and starting replication from the new master 192.168.10.130(192.168.10.130:3306)..

Wed Apr 20 06:04:07 2016 - [info] Executed CHANGE MASTER.

Wed Apr 20 06:04:07 2016 - [info] Slave started.

Wed Apr 20 06:04:07 2016 - [info] All new slave servers switched successfully.

Wed Apr 20 06:04:07 2016 - [info]

Wed Apr 20 06:04:07 2016 - [info] \* Phase 5: New master cleanup phase..

Wed Apr 20 06:04:07 2016 - [info]

Wed Apr 20 06:04:07 2016 - [info] 192.168.10.130: Resetting slave info succeeded.

Wed Apr 20 06:04:07 2016 - [info] Switching master to 192.168.10.130(192.168.10.130:3306) completed successfully.

[root@manager ~]#

7.4 查看从库同步状态

[root@slave01 ~]# mysql -uroot -pmeishi -h 192.168.10.131

mysql> show slave status\G

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. row \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Slave\_IO\_State: Waiting for master to send event

Master\_Host: **192.168.10.130**

Master\_User: rep

Master\_Port: 3306

Connect\_Retry: 60

Master\_Log\_File: mysql-bin.000007

Read\_Master\_Log\_Pos: 107

Relay\_Log\_File: relay-bin.000002

Relay\_Log\_Pos: 253

Relay\_Master\_Log\_File: mysql-bin.000007

Slave\_IO\_Running: Yes

Slave\_SQL\_Running: Yes

Replicate\_Do\_DB:

Replicate\_Ignore\_DB: mysql

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: 107

Relay\_Log\_Space: 403

Until\_Condition: None

Until\_Log\_File:

Until\_Log\_Pos: 0

Master\_SSL\_Allowed: No

Master\_SSL\_CA\_File:

Master\_SSL\_CA\_Path:

Master\_SSL\_Cert:

Master\_SSL\_Cipher:

Master\_SSL\_Key:

Seconds\_Behind\_Master: 0

Master\_SSL\_Verify\_Server\_Cert: No

Last\_IO\_Errno: 0

Last\_IO\_Error:

Last\_SQL\_Errno: 0

Last\_SQL\_Error:

Replicate\_Ignore\_Server\_Ids:

Master\_Server\_Id: 1

1 row in set (0.00 sec)

7.5 小结：

**注意，在线切换的时候应用架构需要考虑以下两个问题：**

1.自动识别master和slave的问题（master的机器可能会切换），如果采用了vip的方式，基本可以解决这个问题。

2.负载均衡的问题（可以定义大概的读写比例，每台机器可承担的负载比例，当有机器离开集群时，需要考虑这个问题）

**为了保证数据完全一致性，在最快的时间内完成切换，MHA的在线切换必须满足以下条件才会切换成功，否则会切换失败。**

**(1)所有slave的IO线程都在运行**

**(2)所有slave的SQL线程都在运行**

**(3)所有的show slave status的输出中Seconds\_Behind\_Master参数小于或者等于running\_updates\_limit秒，如果在切换过程中不指定running\_updates\_limit,那么默认情况下running\_updates\_limit为1秒。**

**(4)在master端，通过show processlist输出，没有一个更新花费的时间大于running\_updates\_limit秒。**