## 环境

centos7

主 172.17.15.61

主（备） 172.17.15.66

mysql：mysql-5.7.21-linux-glibc2.12-x86\_64.tar.gz

keepalived:keepalived-1.3.5.tar.gz

## Mysql安装

mysql安装：（参考https://www.cnblogs.com/dadadechengzi/p/6723686.html）

①卸载系统自带的Mariadb

[root@hdp265dnsnfs ~]# rpm -qa|grep mariadb

mariadb-libs-5.5.44-2.el7.centos.x86\_64

[root@hdp265dnsnfs ~]# rpm -e --nodeps mariadb-libs-5.5.44-2.el7.centos.x86\_64

②删除etc目录下的my.cnf文件

[root@hdp265dnsnfs ~]# rm /etc/my.cnf

rm: cannot remove ?etc/my.cnf? No such file or directory

③检查mysql是否存在

[root@hdp265dnsnfs ~]# rpm -qa | grep mysql

④检查mysql组和用户是否存在，如无创建，创建mysql用户组，创建一个用户名为mysql的用户并加入mysql用户组

[root@hdp265dnsnfs ~]# cat /etc/group | grep mysql

[root@hdp265dnsnfs ~]# cat /etc/passwd | grep mysql

[root@hdp265dnsnfs ~]# groupadd mysql

[root@hdp265dnsnfs ~]# useradd -g mysql mysql

[root@hdp265dnsnfs ~]# passwd mysql

⑤安装mysql 目录/home/mysql

[root@hdp265dnsnfs mysql]# tar -zxvf mysql-5.7.21-linux-glibc2.12-x86\_64.tar.gz

[root@hdp265dnsnfs mysql]# mv mysql-5.7.21-linux-glibc2.12-x86\_64/ mysql57

⑥更改所属的组和用户

[root@hdp265dnsnfs var]# chown -R mysql mysql57/

[root@hdp265dnsnfs var]# chgrp -R mysql mysql57/

[root@hdp265dnsnfs var]# cd mysql57/

[root@hdp265dnsnfs mysql57]# mkdir data

[root@hdp265dnsnfs mysql57]# chown -R mysql:mysql data

⑦在etc下新建配置文件my.cnf，并在该文件内添加以下配置

[root@hdp265dnsnfs mysql57]# vi /etc/my.cnf

[mysql]

# 设置mysql客户端默认字符集

default-character-set=utf8

[mysqld]

skip-name-resolve

#设置3306端口

port = 3306

# 设置mysql的安装目录

basedir=/home/mysql/mysql57

# 设置mysql数据库的数据的存放目录

datadir=/home/mysql/mysql57/data

# 允许最大连接数

max\_connections=200

# 服务端使用的字符集默认为8比特编码的latin1字符集

character-set-server=utf8

# 创建新表时将使用的默认存储引擎

default-storage-engine=INNODB

lower\_case\_table\_names=1

max\_allowed\_packet=16M

⑧安装和初始化

[root@hdp265dnsnfs mysql57]# bin/mysql\_install\_db --user=mysql --basedir=/home/mysql/mysql57/ --datadir=/home/mysql/mysql57/data/

2017-04-17 17:40:02 [WARNING] mysql\_install\_db is deprecated. Please consider switching to mysqld --initialize

2017-04-17 17:40:05 [WARNING] The bootstrap log isn't empty:

2017-04-17 17:40:05 [WARNING] 2017-04-17T09:40:02.728710Z 0 [Warning] --bootstrap is deprecated. Please consider using --initialize instead

2017-04-17T09:40:02.729161Z 0 [Warning] Changed limits: max\_open\_files: 1024 (requested 5000)

2017-04-17T09:40:02.729167Z 0 [Warning] Changed limits: table\_open\_cache: 407 (requested 2000)

[root@hdp265dnsnfs mysql57]# cp ./support-files/mysql.server /etc/init.d/mysqld

[root@hdp265dnsnfs mysql57]# chown 777 /etc/my.cnf

[root@hdp265dnsnfs mysql57]# chmod +x /etc/init.d/mysqld

[root@hdp265dnsnfs mysql57]# /etc/init.d/mysqld restart

Shutting down MySQL.. SUCCESS!

Starting MySQL. SUCCESS!

⑨设置开机启动及环境变量

[root@hdp265dnsnfs mysql57]# chkconfig --level 35 mysqld on

[root@hdp265dnsnfs mysql57]# chkconfig --list mysqld

[root@hdp265dnsnfs mysql57]# chmod +x /etc/rc.d/init.d/mysqld

[root@hdp265dnsnfs mysql57]# chkconfig --add mysqld

[root@hdp265dnsnfs mysql57]# chkconfig --list mysqld

[root@hdp265dnsnfs mysql57]# service mysqld status

SUCCESS! MySQL running (4475)

[root@hdp265dnsnfs mysql57]# vi /etc/profile

export PATH=$PATH:/var/mysql57/bin

[root@hdp265dnsnfs mysql57]# source /etc/profile

⑩初始化mysql登录

[root@hdp265dnsnfs bin]# cat /root/.mysql\_secret

# Password set for user 'root@localhost' at 2017-04-17 17:40:02

\_pB\*3VZl5T<6

[root@hdp265dnsnfs bin]# mysql -uroot -p

Enter password:

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 5

..........

mysql> set PASSWORD = PASSWORD('111111');

mysql> flush privileges;

mysql> grant all privileges on \*.\* to root@'%' identified by '123456';

mysql> flush privileges;

⑪关闭防火墙

## Mysql双主准备（SSH）

在172.17.15.61和172.17.15.66上分别执行：

[root@hdp265dnsnfs ~]# ssh-keygen -t rsa

[root@hdp265dnsnfs ~]# ssh-copy-id -i /root/.ssh/id\_rsa.pub root@172.17.15.61

[root@hdp265dnsnfs ~]# ssh-copy-id -i /root/.ssh/id\_rsa.pub root@172.17.15.66

## Mysql双主

mysql双主复制环境配置（参考：https://www.cnblogs.com/phpstudy2015-6/p/6485819.html）

安装配置双主环境前，要先保证两个mysql环境里的数据一致

①分别修改两个服务器的/etc/my.cnf文件

修改172.17.15.61

[root@hdp265dnsnfs ~]# vi /etc/my.cnf

[mysql]

# 设置mysql客户端默认字符集

default-character-set=utf8

[mysqld]

skip-name-resolve

#设置3306端口

port = 3306

# 设置mysql的安装目录

basedir=/home/mysql/mysql57

# 设置mysql数据库的数据的存放目录

datadir=/home/mysql/mysql57/data

# 允许最大连接数

max\_connections=200

# 服务端使用的字符集默认为8比特编码的latin1字符集

character-set-server=utf8

# 创建新表时将使用的默认存储引擎

default-storage-engine=INNODB

lower\_case\_table\_names=1

max\_allowed\_packet=16M

log-bin=mysql-bin

server-id=1

auto\_increment\_increment=2

auto\_increment\_offset=1

#需要同步的数据库，测试时用的test

replicate-do-db=test

binlog-ignore-db = mysql

binlog-ignore-db = information\_schema

log-slave-updates

symbolic-links=0

修改172.17.15.66

[root@hdp265dnsnfs ~]# vi /etc/my.cnf

[mysql]

# 设置mysql客户端默认字符集

default-character-set=utf8

[mysqld]

skip-name-resolve

#设置3306端口

port = 3306

# 设置mysql的安装目录

basedir=/home/mysql/mysql57

# 设置mysql数据库的数据的存放目录

datadir=/home/mysql/mysql57/data

# 允许最大连接数

max\_connections=200

# 服务端使用的字符集默认为8比特编码的latin1字符集

character-set-server=utf8

# 创建新表时将使用的默认存储引擎

default-storage-engine=INNODB

lower\_case\_table\_names=1

max\_allowed\_packet=16M

log-bin=mysql-bin

server-id=2

auto\_increment\_increment=2

auto\_increment\_offset=2

#需要同步的数据，测试时用的test

replicate-do-db=test

binlog-ignore-db = mysql

binlog-ignore-db = information\_schema

log-slave-updates

symbolic-links=0

②在172.17.15.61上，登录mysql，执行：

mysql>stop slave;

mysql>GRANT REPLICATION SLAVE ON \*.\* TO 'root'@'172.17.15.66' IDENTIFIED BY '123456';

mysql>grant all on \*.\* to 'root'@'172.17.15.66' identified by '123456';

mysql>FLUSH PRIVILEGES;

mysql>show master status \G

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

File: mysql-bin.000001

Position: 654

Binlog\_Do\_DB:

Binlog\_Ignore\_DB: mysql,information\_schema

Executed\_Gtid\_Set:

1 row in set (0.00 sec)

③在172.17.15.66上，登录mysql，执行：

mysql>stop slave;

mysql>GRANT REPLICATION SLAVE ON \*.\* TO 'root'@'172.17.15.61' IDENTIFIED BY '123456';

mysql>grant all on \*.\* to 'root'@'172.17.15.61' identified by '123456';

mysql>FLUSH PRIVILEGES;

mysql>show master status \G

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

File: mysql-bin.000003

Position: 141

Binlog\_Do\_DB:

Binlog\_Ignore\_DB: mysql,information\_schema

Executed\_Gtid\_Set:

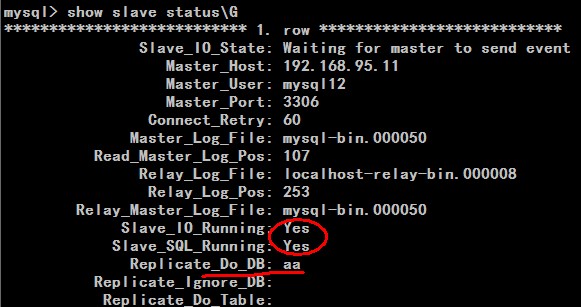
1 row in set (0.00 sec)

④在172.17.15.61上执行

mysql>CHANGE MASTER TO MASTER\_HOST='172.17.15.66', MASTER\_USER='root', MASTER\_PASSWORD='123456', MASTER\_LOG\_FILE='mysql-bin.000003', MASTER\_LOG\_POS=141;

mysql>start slave;

mysql>show slave status \G

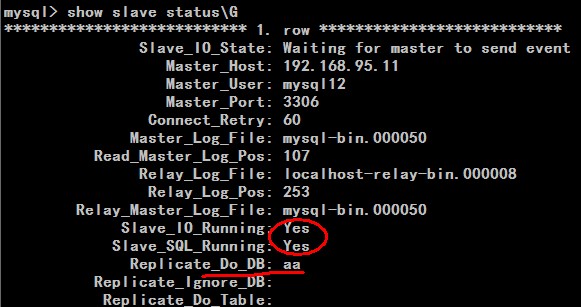


⑤在172.17.15.66上执行

mysql>CHANGE MASTER TO MASTER\_HOST='172.17.15.61', MASTER\_USER='root', MASTER\_PASSWORD='123456', MASTER\_LOG\_FILE='mysql-bin.000001', MASTER\_LOG\_POS=654;

mysql>start slave;

mysql>show slave status \G



⑥测试

⑦注意

 1、主主复制配置文件中auto\_increment\_increment和auto\_increment\_offset只能保证主键不重复，却不能保证主键有序。

2、当配置完成Slave\_IO\_Running、Slave\_SQL\_Running不全为YES时，show slave status\G信息中有错误提示，可根据错误提示进行更正。

3、Slave\_IO\_Running、Slave\_SQL\_Running不全为YES时，大多数问题都是数据不统一导致。

常见出错点：

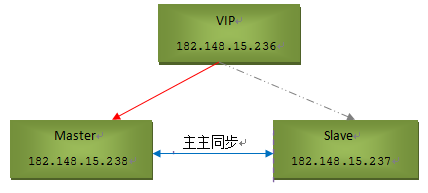
1、两台数据库都存在db数据库，而第一台MySQL db中有tab1，第二台MySQL db中没有tab1，那肯定不能成功。

2、已经获取了数据的二进制日志名和位置，又进行了数据操作，导致POS发生变更。在配置CHANGE MASTER时还是用到之前的POS。

3、stop slave后，数据变更，再start slave。出错。

终极更正法：重新执行一遍CHANGE MASTER就好了。

## keepalived安装部署+mysql双主热备



keepalived环境配置

Centos7 版本

Master1：172.17.15.61        安装mysql和keepalived

Master2：172.17.15.66        安装mysql和keepalived

VIP：172.17.15.218

①安装keepalived并将其配置成系统服务。master1和master2两台机器上同样进行如下操作：

[root@master1 ~]# yum install -y openssl-devel

[root@master1 ~]# cd /home/keepalived/

[root@master1 keepalived]# wget http://www.keepalived.org/software/keepalived-1.3.5.tar.gz

[root@master1 keepalived]# tar -zvxf keepalived-1.3.5.tar.gz

[root@master1 keepalived]# cd keepalived-1.3.5

[root@master1 keepalived-1.3.5]# ./configure --prefix=/usr/local/keepalived

[root@master1 keepalived-1.3.5]# make && make install

[root@master1 keepalived-1.3.5]# cp /home/keepalived/keepalived-1.3.5/keepalived/etc/init.d/keepalived /etc/rc.d/init.d/

[root@master1 keepalived-1.3.5]# cp /usr/local/keepalived/etc/sysconfig/keepalived /etc/sysconfig/

[root@master1 keepalived-1.3.5]# mkdir /etc/keepalived/

[root@master1 keepalived-1.3.5]# cp /usr/local/keepalived/etc/keepalived/keepalived.conf /etc/keepalived/

[root@master1 keepalived-1.3.5]# cp /usr/local/keepalived/sbin/keepalived /usr/sbin/

[root@master1 keepalived-1.3.5]# echo "/etc/init.d/keepalived start" >> /etc/rc.local

②master1机器上的keepalived.conf配置（下面配置中没有使用lvs的负载均衡功能，所以不需要配置虚拟服务器virtual server）

[root@master1 ~]# cp /etc/keepalived/keepalived.conf /etc/keepalived/keepalived.conf.bak

[root@master1 ~]# vi /etc/keepalived/keepalived.conf

（＃清空默认内容，直接采用下面配置：）

! Configuration File for keepalived

global\_defs {

notification\_email {

gao.zhongke@bohui.com.cn

}

notification\_email\_from gao.zhongke@bohui.com.cn

smtp\_server 127.0.0.1

smtp\_connect\_timeout 30

router\_id MASTER-HA

}

#检测mysql服务是否在运行。有很多方式，比如进程，用脚本检测等等

vrrp\_script chk\_mysql\_port {

#这里通过脚本监测

script "/opt/chk\_mysql.sh"

#脚本执行间隔，每2s检测一次

interval 2

#脚本结果导致的优先级变更，检测失败（脚本返回非0）则优先级 -5

weight -5

#检测连续2次失败才算确定是真失败。会用weight减少优先级（1-255之间）

fall 2

#检测1次成功就算成功。但不修改优先级

rise 1

}

vrrp\_instance VI\_1 {

#状态MASTER或BACKUP

state MASTER

#指定虚拟ip的网卡接口

interface eno16777984

#使用这个地址作为多播包的源IP，而不是使用interface eth0上的IP

mcast\_src\_ip 172.17.15.61

#路由器标识，MASTER和BACKUP必须是一致的

virtual\_router\_id 51

#定义优先级，数字越大，优先级越高

priority 101

advert\_int 1

authentication {

auth\_type PASS

auth\_pass Hello123

}

virtual\_ipaddress {

#虚拟IP为172.17.15.218

172.17.15.218

}

track\_script {

chk\_mysql\_port

}

}

③编写切换脚本。KeepAlived做心跳检测，如果Master的MySQL服务挂了(3306端口挂了),那么它就会选择自杀。Slave的KeepAlived通过心跳检测发现这个情况，就会将VIP的请求接管

[root@master1 ~]# vim /opt/chk\_mysql.sh

#!/bin/bash

counter=$(netstat -na|grep "LISTEN"|grep "3306"|wc -l)

if [ "${counter}" -eq 0 ]; then

    /etc/init.d/keepalived stop

fi

[root@master1 ~]# chmod 755 /opt/chk\_mysql.sh

④master2机器上的keepalived配置。master2机器上的keepalived.conf文件只修改priority为90、nopreempt不设置、real\_server设置本地IP。

[root@master2 ~]# cp /etc/keepalived/keepalived.conf /etc/keepalived/keepalived.conf.bak

[root@master2 ~]# vi /etc/keepalived/keepalived.conf

! Configuration File for keepalived

global\_defs {

notification\_email {

gao.zhongke@bohui.com.cn

}

notification\_email\_from gao.zhongke@bohui.com.cn

smtp\_server 127.0.0.1

smtp\_connect\_timeout 30

router\_id MASTER-HA

}

vrrp\_script chk\_mysql\_port {

script "/opt/chk\_mysql.sh"

interval 2

weight -5

fall 2

rise 1

}

vrrp\_instance VI\_1 {

state BACKUP

interface eno16777984

mcast\_src\_ip 172.17.15.66

virtual\_router\_id 51

priority 99

advert\_int 1

authentication {

auth\_type PASS

auth\_pass Hello123

}

virtual\_ipaddress {

172.17.15.218

}

track\_script {

chk\_mysql\_port

}

}

[root@master2 ~]# cat /opt/chk\_mysql.sh

#!/bin/bash

counter=$(netstat -na|grep "LISTEN"|grep "3306"|wc -l)

if [ "${counter}" -eq 0 ]; then

    /etc/init.d/keepalived stop

fi

[root@master2 ~]# chmod 755 /opt/chk\_mysql.sh

⑤在master1和master2两台机器修改PIDFILE

[root@master1 ~]# vi /lib/systemd/system/keepalived.service

#修改

PIDFile=/var/run/keepalived.pid

[root@master1 ~]# systemctl daemon-reload

[root@master2 ~]# vi /lib/systemd/system/keepalived.service

#修改

PIDFile=/var/run/keepalived.pid

[root@master2 ~]# systemctl daemon-reload

⑥分别启动master1、master2里keepalived服务

[root@master1 ~]# /etc/init.d/keepalived start

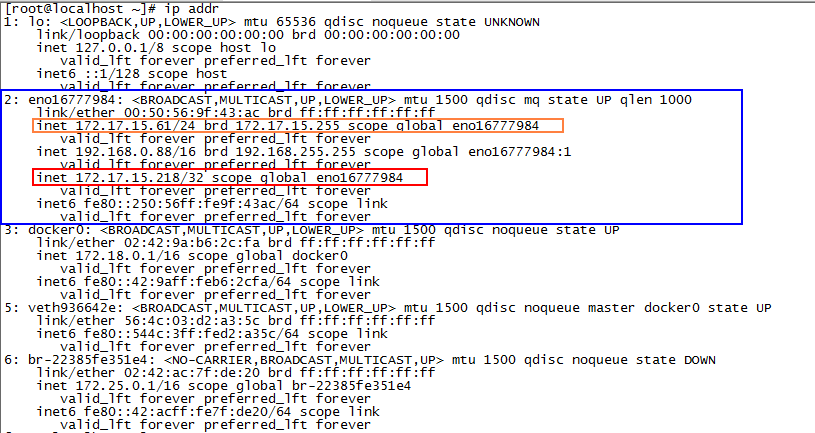
正在启动 keepalived：                                      [确定]

[root@master2 ~]# /etc/init.d/keepalived start

正在启动 keepalived：                                      [确定]

⑦查看测试：master1 是主机 ，master2 是备机

[root@master1 ~]# ip addr



[root@master2 ~]# ip addr

