# Hearbeat 介绍

### 作用：

作用：可以将资源（IP及程序）从一台故障的计算机快速转移到另一台正常机器，称之高可用。

工作原理：

### 工作原理

在配置文件里，可以指定那一台服务器做主，通过监听主服务器的心跳信息，如果在指定时间内没监听心跳，就会取得主服务的资源，代替主服务，时间一般是5-20s。

## 裂脑

服务器之间无法检测到对方的心跳各自启动故障转移，导致同一个VIP同时在服务两端，写入数据同时写入的现象

### 导致裂脑的原因：

心跳线故障（网卡，网线断了，交换机故障）

防火墙阻挡

### Heartbeat IP地址接管和故障转移

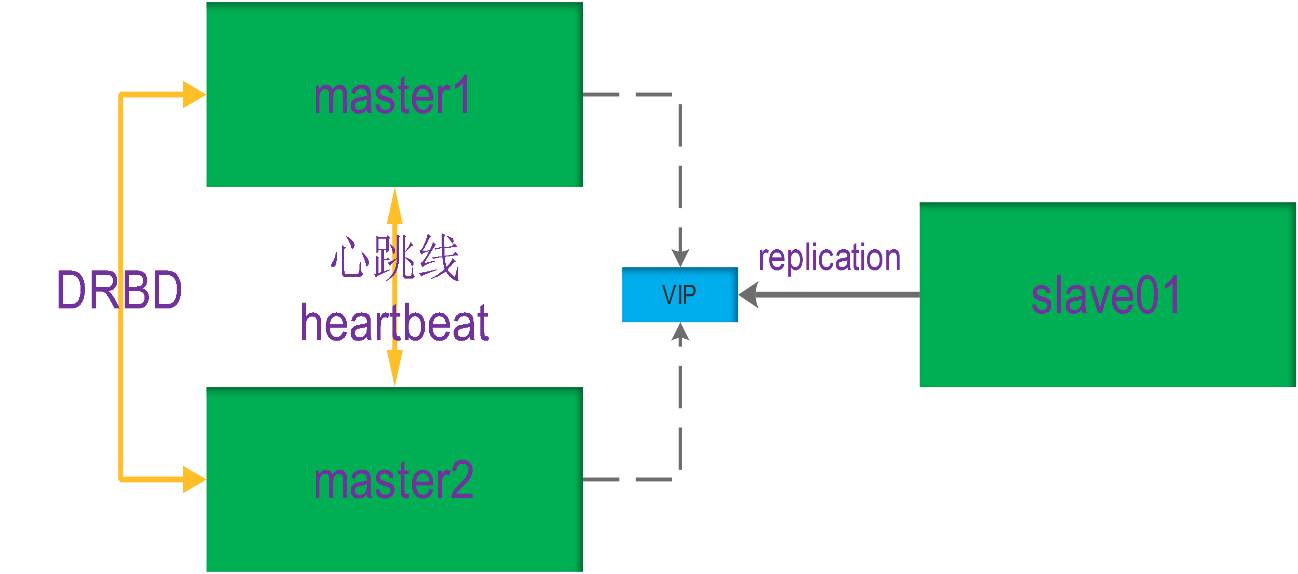
通过IP地址接管和ARP广播进行故障转移。

ARP广播：主故障，备接管资源后，会立即强制更新所有客户端本地的ARP表（清除客户端本地缓存的失败服务器VIP和mac的解析记录），使得客户端和新的主对话。

手工配置VIP的方法

ip addr add 10.0.0.100/24 broadcast 10.0.15.255 dev eth0

### Heartbeat部署图



### 配置主机名:

master1

master2

slave01

### 安装网卡vip 172.16.1.50

master1:

eth0:10.0.0.51(数据库无公网地址)

eth1:172.16.1.51/24(内网)

eth2:192.16.1.51/24(心跳线)

eth3:192.168.1.51/24(DRBD千兆数据传输)

master2:

eth0:10.0.0.52(数据库无公网地址)

eth1:172.16.1.52/24(内网)

eth2:192.16.1.52/24(心跳线)

eth3:192.168.1.52/24(DRBD千兆数据传输)

### 检查：

ping 10.0.0.52

ping 172.16.1.52

ping 192.168.1.52

ping 192.16.1.52

## heartbeat安装：

注意：这里安装用的是6.7的版本

#### 添加路由：（两个都要添加）

[root@master1 tools]# /sbin/route add -host 192.16.1.52 dev eth2

[root@master1 tools]# route -n

Kernel IP routing table

Destination Gateway Genmask Flags Metric Ref Use Iface

192.16.1.52 0.0.0.0 255.255.255.255 UH 0 0 0 eth2

[root@master1 tools]# echo '/sbin/route add -host 192.16.1.52 dev eth2'>>/etc/rc.local

#### 下载软件（两台主安装及配置都一样）

系统版本：5 是2.0版本 6是3.0

wget -O /etc/yum.repos.d/epel.repo http://mirrors.aliyun.com/repo/epel-6.repo

yum install heartbeat -y

[root@master2 ha.d]# rpm -qa heartbeat

heartbeat-3.0.4-2.el6.x86\_64

[root@master1 ~]# ll /usr/share/doc/heartbeat-3.0.4/

总用量 144

-rw-r--r-- 1 root root 1873 12月 3 2013 apphbd.cf

-rw-r--r-- 1 root root 645 12月 3 2013 authkeys

-rw-r--r-- 1 root root 3701 12月 3 2013 AUTHORS

-rw-r--r-- 1 root root 58752 12月 3 2013 ChangeLog

-rw-r--r-- 1 root root 17989 12月 3 2013 COPYING

-rw-r--r-- 1 root root 26532 12月 3 2013 COPYING.LGPL

-rw-r--r-- 1 root root 10502 12月 3 2013 ha.cf

-rw-r--r-- 1 root root 5905 12月 3 2013 haresources

-rw-r--r-- 1 root root 2935 12月 3 2013 README

[root@master1 ~]# ll /etc/ha.d/

总用量 20

-rwxr-xr-x 1 root root 745 12月 3 2013 harc

drwxr-xr-x 2 root root 4096 12月 27 14:54 rc.d

-rw-r--r-- 1 root root 692 12月 3 2013 README.config

drwxr-xr-x 2 root root 4096 12月 27 14:54 resource.d

-rw-r--r-- 1 root root 2082 11月 10 19:00 shellfuncs

#### 对三个配置文件配置：

ha.cf文件：

[root@master1 ~]# cat /etc/ha.d/ha.cf

#log configure

debugfile /var/log/ha-debug

logfile /var/log/ha-log

logfacility local1

#options configure

keepalive 2

deadtime 30

warntime 10

initdead 120

#bcast eth2

mcast eth1 225.0.0.51 694 1 0 #采用多播方式

#node configure

auto\_failback on

node master1 #主节点主机名

node master2 #备节点主机名

crm no

ha.cf 详细配置说明：

debugfile /var/log/ha-debug #调试日志

logfile /var/log/ha-log #日志位置

logfacility local1 在syslog服务中配置通过locall设备接收日志

#options configure

keepalive 2 心跳间隔 广播发送间隔

deadtime 30 30s没接到主的信号，接管

warntime 10 10秒没接到信号，就写一个警告到日志

initdead 120 HB运行后，等待120秒启动资源，

#bcast eth2

mcast eth1 225.0.0.51 694 1 0 #采用多播方式 694为默认的端口号

#node configure

auto\_failback on 自动切回服务

node master1 #主节点主机名

node master2 #备节点主机名

crm no 是否开启群资源管理功能

authkeys文件：

[root@master1 ~]# cat /etc/ha.d/authkeys

auth 1

1 sha1 47e9336850f1db6fa58bc470bc9b7810eb397f04

haresources文件：

[root@master1 ~]# cat /etc/ha.d/haresources

master1 IPaddr::172.16.1.50/24/eth1

#master1 IPaddr::192.168.4.1/16/eth1 drbddisk::data Filesystem::/dev/drbd1::/data::ext3 mysqld

master2 IPaddr::172.16.1.60/24/eth1

把配置好的文件推到master1

[root@master1 ha.d]# scp -P52113 -rp ha.cf authkeys haresources root@172.16.1.52:/etc/ha.d/

启动：

[root@master1 ha.d]# chmod 600 authkeys

[root@master1 ha.d]# /etc/init.d/heartbeat start

[root@master1 ha.d]# ip addr |grep 172.16.1.

inet 172.16.1.51/24 brd 172.16.1.255 scope global eth1

inet 172.16.1.50/24 brd 172.16.1.255 scope global secondary eth1

查看日志：

cat /var/log/ha-log

cat /var/log/ha-debug 命令的执行过程

cat /var/log/messages

#### 报错问题：

1）没设置密码600的权限

[root@master1 ha.d]# /etc/init.d/heartbeat restart Stopping High-Availability services: Done.

Waiting to allow resource takeover to complete:Done.

Starting High-Availability services: INFO: Resource is stopped

INFO: Resource is stopped

Heartbeat failure [rc=6]. Failed.

heartbeat[2649]: 2015/12/27\_15:04:58 info: Pacemaker support: no

heartbeat[2649]: 2015/12/27\_15:04:58 ERROR: Bad permissions on keyfile [/etc/ha.d//authkeys], 600 recommended.

heartbeat[2649]: 2015/12/27\_15:04:58 ERROR: Authentication configuration error.

heartbeat[2649]: 2015/12/27\_15:04:58 ERROR: Configuration error, heartbeat not started

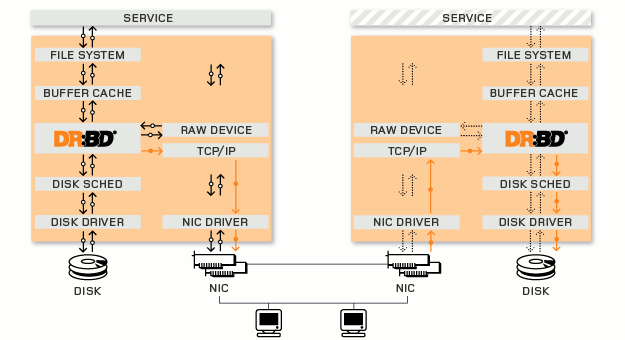
#### 2）列脑 ：

防火墙影响

# DRBD介绍

### 初步了解

这个软件可以实现在网络中两台服务器之间基于块设备级别的实时或异步镜像或同步复制，drbd是基于文件系统底层的，rsync+inotify是文件系统之上的实际物理文件同步，drbd效率更高，效果更好。



http://drbd.linbit.com/

### drbd有实时和异步两种模式

#### 实时同步：

数据写入到本地磁盘和远端所有磁盘都成功才返回成功写入，drbd服务的协议C级别就是这种同步模式，防止本地和远端的数据丢失和不一致，生产常用。

#### 异步同步：

不管远端是否成功，本地成功就返回成功。

还有可能是数据写入到本地服务器或远端的buffer成功后，返回成功，这是协议A、B级别。

### DRBD的3种同步复制协议

协议A：异步协议。本地成功后立即返回，数据放在发送的buffer中，可能丢失。

协议B：内存同步（半同步）协议，本地成功并将数据发送到对方后立即返回，如果此时宕机可能丢失。

协议C：同步协议，本地和对方都写入成功确认返回成功，数据不丢失。

工作中一班用协议C

## drbd的安装部署：http://drbd.linbit.com/

#### 配置：数据传输路由（两个都配）

[root@master1 ha.d]# /sbin/route add -host 192.168.1.52 dev eth3

echo '/sbin/route add -host 192.168.1.52 dev eth3' >>/etc/rc.local

[root@master1 ha.d]# route -n

Kernel IP routing table

Destination Gateway Genmask Flags Metric Ref Use Iface

192.168.1.52 0.0.0.0 255.255.255.255 UH 0 0 0 eth3

192.16.1.52 0.0.0.0 255.255.255.255 UH 0 0 0 eth2

[root@master1 ha.d]# tail -2 /etc/rc.local

/sbin/route add -host 192.16.1.52 dev eth2

/sbin/route add -host 192.168.1.52 dev eth3

#### 两台机器都增加磁盘

[root@master1 ha.d]# reboot

[root@master1 ~]# fdisk -l

Disk /dev/sdb: 1073 MB, 1073741824 bytes

255 heads, 63 sectors/track, 130 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk identifier: 0x00000000

#### 说明：sdb磁盘分两个分区sdb1和sdb2

[root@master1 ~]# fdisk /dev/sdb

Device contains neither a valid DOS partition table, nor Sun, SGI or OSF disklabel

Building a new DOS disklabel with disk identifier 0xfe6036cd.

Changes will remain in memory only, until you decide to write them.

After that, of course, the previous content won't be recoverable.

Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)

WARNING: DOS-compatible mode is deprecated. It's strongly recommended to

switch off the mode (command 'c') and change display units to

sectors (command 'u').

Command (m for help): p

Disk /dev/sdb: 1073 MB, 1073741824 bytes

255 heads, 63 sectors/track, 130 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk identifier: 0xfe6036cd

Device Boot Start End Blocks Id System

Command (m for help): m

Command action

a toggle a bootable flag

b edit bsd disklabel

c toggle the dos compatibility flag

d delete a partition

l list known partition types

m print this menu

n add a new partition

o create a new empty DOS partition table

p print the partition table

q quit without saving changes

s create a new empty Sun disklabel

t change a partition's system id

u change display/entry units

v verify the partition table

w write table to disk and exit

x extra functionality (experts only)

Command (m for help): n

Command action

e extended

p primary partition (1-4)

p

Partition number (1-4): 1

First cylinder (1-130, default 1):

Using default value 1

Last cylinder, +cylinders or +size{K,M,G} (1-130, default 130): +768M

Command (m for help): n

Command action

e extended

p primary partition (1-4)

p

Partition number (1-4): 2

First cylinder (100-130, default 100):

Using default value 100

Last cylinder, +cylinders or +size{K,M,G} (100-130, default 130):

Using default value 130

Command (m for help): w

The partition table has been altered!

Calling ioctl() to re-read partition table.

Syncing disks.

[root@master1 ~]# fdisk -l

Device Boot Start End Blocks Id System

/dev/sdb1 1 99 795186 83 Linux

/dev/sdb2 100 130 249007+ 83 Linux

格式化分区：

[root@master1 ~]# partprobe /dev/sdb 通知分区

[root@master1 ~]# mkfs.ext4 /dev/sdb1

mke2fs 1.41.12 (17-May-2010)

文件系统标签=

操作系统:Linux

块大小=4096 (log=2)

分块大小=4096 (log=2)

Stride=0 blocks, Stripe width=0 blocks

49728 inodes, 198796 blocks

9939 blocks (5.00%) reserved for the super user

第一个数据块=0

Maximum filesystem blocks=205520896

7 block groups

32768 blocks per group, 32768 fragments per group

7104 inodes per group

Superblock backups stored on blocks:

32768, 98304, 163840

正在写入inode表: 完成

Creating journal (4096 blocks): 完成

Writing superblocks and filesystem accounting information: 完成

This filesystem will be automatically checked every 22 mounts or

180 days, whichever comes first. Use tune2fs -c or -i to override.

说明：sdb2分区为meta data分区，不需要格式化操作

[root@master1 ~]# tune2fs -c -1 /dev/sdb1 不让检查

说明：设置最大挂载数为-1

#### 安装drbd

[root@master1 ~]# cd /etc/yum.repos.d/

[root@master1 yum.repos.d]# wget http://elrepo.reloumirrors.net/elrepo/el6/i386/RPMS/elrepo-release-6-4.el6.elrepo.noarch.rpm

[root@master1 yum.repos.d]# rpm -ivh elrepo-release-6-4.el6.elrepo.noarch.rpm

warning: elrepo-release-6-4.el6.elrepo.noarch.rpm: Header V4 DSA/SHA1 Signature, key ID baadae52: NOKEY

Preparing... ########################################### [100%]

1:elrepo-release ########################################### [100%]

这是163的源：（可以不下）

[root@master1 yum.repos.d]# wget http://mirrors.163.com/.help/CentOS6-Base-163.repo

[root@master1 yum.repos.d]# yum install kmod-drbd83 drbd83 -y 可以用8.4版本

[root@master1 yum.repos.d]# lsmod |grep drbd

[root@master1 yum.repos.d]# modprobe drbd 内核加载drbd 可以放到rc.local下

[root@master1 yum.repos.d]# lsmod |grep drbd

drbd 332493 0

[root@master1 yum.repos.d]# cp /etc/drbd.conf /etc/drbd.conf.ori

[root@master1 ha.d]# less /etc/drbd.d/global\_common.conf 模板文件

[root@master1 etc]# cd cd /etc/

[root@master1 etc]# rz -y

[root@master1 ~]# vim /etc/drbd.conf

# You can find an example in /usr/share/doc/drbd.../drbd.conf.example

#include "drbd.d/global\_common.conf";

#include "drbd.d/\*.res";

##############################

global {

minor-count 64;

# dialog-refresh 5; # 5 seconds

# disable-ip-verification;

usage-count no;

}

common {

protocol C;

disk {

on-io-error detach;

#size 454G;

no-disk-flushes;

no-md-flushes;

}

net {

sndbuf-size 512k;

# timeout 60; # 6 seconds (unit = 0.1 seconds)

# connect-int 10; # 10 seconds (unit = 1 second)

# ping-int 10; # 10 seconds (unit = 1 second)

# ping-timeout 5; # 500 ms (unit = 0.1 seconds)

max-buffers 8000;

unplug-watermark 1024;

max-epoch-size 8000;

# ko-count 4;

# allow-two-primaries;

cram-hmac-alg "sha1";

shared-secret "hdhwXes23sYEhart8t";

after-sb-0pri disconnect;

after-sb-1pri disconnect;

after-sb-2pri disconnect;

rr-conflict disconnect;

# data-integrity-alg "md5";

# no-tcp-cork;

}

syncer {

rate 120M;

al-extents 517;

}

}

resource data {

on master1 {

device /dev/drbd0;

disk /dev/sdb1;

address 192.168.1.51:7788;

meta-disk /dev/sdb2[0];

}

on master2 {

device /dev/drbd0;

disk /dev/sdb1;

address 192.168.1.52:7788;

meta-disk /dev/sdb2[0];

}

}

#### 初始化data

[root@master1 etc]# drbdadm create-md data

Writing meta data...

initializing activity log

NOT initialized bitmap

New drbd meta data block successfully created.

这里如果报错：有可能分区没写入分区表，需要重启一下，或重新分区并重启

#### 启动：

[root@master1 etc]# drbdadm up data

或者：drbdadm up all

查看：

只启动一个：

[root@master1 ha.d]# cat /proc/drbd

version: 8.3.16 (api:88/proto:86-97)

GIT-hash: a798fa7e274428a357657fb52f0ecf40192c1985 build by phil@Build64R6, 2014-11-24 14:51:37

0: cs:WFConnection ro:Secondary/Unknown ds:Inconsistent/DUnknown C r----s

ns:0 nr:0 dw:0 dr:0 al:0 bm:0 lo:0 pe:0 ua:0 ap:0 ep:1 wo:d oos:522084

启动两个：

[root@master2 ~]# cat /proc/drbd

version: 8.3.16 (api:88/proto:86-97)

GIT-hash: a798fa7e274428a357657fb52f0ecf40192c1985 build by phil@Build64R6, 2014-11-24 14:51:37

0: cs:Connected ro:Secondary/Secondary ds:Inconsistent/Inconsistent C r-----

ns:0 nr:0 dw:0 dr:0 al:0 bm:0 lo:0 pe:0 ua:0 ap:0 ep:1 wo:d oos:522084

1: cs:WFConnection ro:Secondary/Unknown ds:Diskless/DUnknown C r-----

ns:0 nr:0 dw:0 dr:0 al:0 bm:0 lo:0 pe:0 ua:0 ap:0 ep:1 wo:b oos:0

让本地成为主，并同步数据的命令

[root@master1 etc]# drbdadm -- --overwrite-data-of-peer primary data

ns:63372 nr:0 dw:0 dr:67346 al:0 bm:3 lo:0 pe:2 ua:26 ap:0 ep:1 wo:d oos:458852

[=>..................] sync'ed: 12.5% (458852/522084)K

0: cs:Connected ro:Primary/Secondary ds:UpToDate/UpToDate C r-----

从的内容

ns:0 nr:459776 dw:459776 dr:0 al:0 bm:28 lo:0 pe:66 ua:0 ap:0 ep:1 wo:d oos:62308

[================>...] sync'ed: 88.3% (62308/522084)K

0: cs:Connected ro:Secondary/Primary ds:UpToDate/UpToDate C r-----

主变从的命令:

[root@master1 3306]# /usr/share/heartbeat/hb\_standby

主的挂载：

[root@master1 etc]# mount /dev/drbd0 /data/

测试：

主：

[root@master1 etc]# cd /data/

[root@master1 data]# ll

总用量 16

drwx------ 2 root root 16384 12月 27 15:28 lost+found

[root@master1 data]# touch 123

[root@master1 data]# ll

总用量 16

-rw-r--r-- 1 root root 0 12月 27 16:01 123

drwx------ 2 root root 16384 12月 27 15:28 lost+found

从：

[root@master2 etc]# drbdadm down data

[root@master2 etc]# mount /dev/sdb1 /mnt/

[root@master2 etc]# cd /mnt/

[root@master2 mnt]# ll

总用量 16

-rw-r--r-- 1 root root 0 12月 27 16:01 123

drwx------ 2 root root 16384 12月 27 15:28 lost+found

测试成功

从的挂载问题：

[root@master2 data]# mount /dev/drbd0 /data/

mount: you must specify the filesystem type

[root@master2 data]# mount /dev/sdb1 /data/

mount: /dev/sdb1 already mounted or /data/ busy

只打开一个drbd会出现：（两个都开启就没有端口了）

[root@master1 data]# ss -lntup|grep 7788

tcp LISTEN 0 5 192.168.1.51:7788 \*:\*

#### 问题小结：

列脑可能的原因：

chkconfig heartbeat off

chkconfig drbd off

echo "/etc/init.d/heartbeat start" >>/etc/rc.local

# MySQL安装

在master1安装MySQL：

[root@master1 data]# yum install ncurses-devel libaio-devel cmake -y

[root@master1 data]# useradd mysql -s /sbin/nologin -M

[root@master1 data]# cd /home/oldboy/tools/

[root@master1 tools]# rz -y

[root@master1 tools]# tar xf mysql-5.5.32.tar.gz

[root@master1 tools]# cd mysql-5.5.32

[root@master1 mysql-5.5.32]# cmake . -DCMAKE\_INSTALL\_PREFIX=/application/mysql-5.5.32 \

> -DMYSQL\_DATADIR=/application/mysql-5.5.32/data \

> -DMYSQL\_UNIX\_ADDR=/application/mysql-5.5.32/tmp/mysql.sock \

> -DDEFAULT\_CHARSET=utf8 \

> -DDEFAULT\_COLLATION=utf8\_general\_ci \

> -DEXTRA\_CHARSETS=gbk,gb2312,utf8,ascii \

> -DENABLED\_LOCAL\_INFILE=ON \

> -DWITH\_INNOBASE\_STORAGE\_ENGINE=1 \

> -DWITH\_FEDERATED\_STORAGE\_ENGINE=1 \

> -DWITH\_BLACKHOLE\_STORAGE\_ENGINE=1 \

> -DWITHOUT\_EXAMPLE\_STORAGE\_ENGINE=1 \

> -DWITHOUT\_PARTITION\_STORAGE\_ENGINE=1 \

> -DWITH\_FAST\_MUTEXES=1 \

> -DWITH\_ZLIB=bundled \

> -DENABLED\_LOCAL\_INFILE=1 \

> -DWITH\_READLINE=1 \

> -DWITH\_EMBEDDED\_SERVER=1 \

> -DWITH\_DEBUG=0

make

make install

ln -s /application/mysql-5.5.32/ /application/mysql

mkdir -p /data/{3306,3307}/data

cd /

rz data.zip

unzip data.zip

cd /data

find /data -type f -name "mysql"|xargs chmod +x

chown -R mysql.mysql /data/

cd /application/mysql/scripts/

./mysql\_install\_db --basedir=/application/mysql/ --datadir=/data/3306/data --user=mysql

./mysql\_install\_db --basedir=/application/mysql/ --datadir=/data/3307/data --user=mysql

/data/3306/mysql start

/data/3307/mysql start

netstat -lntup|grep 330

cp /application/mysql/bin/\* /usr/local/sbin/

mysqladmin password oldboy123 -S /data/3306/mysql.sock

mysqladmin password oldboy456 -S /data/3307/mysql.sock

mysql -uroot -poldboy123 -S /data/3306/mysql.sock

mysql -uroot -poldboy456 -S /data/3307/mysql.sock

把MySQL拷贝到master2和slave01

master1:（master2不用拷贝/data/只需拷贝/application/）

scp -P52113 -rp /application/\* root@172.16.1.41:/application/

scp -P52113 -rp /data/\* root@172.16.1.41:/data/

在master2和slave01

useradd mysql -s /sbin/nologin -M

find /data -type f -name "mysql"|xargs chmod +x

chown -R mysql.mysql /data/

cp /application/mysql/bin/\* /usr/local/sbin/

安装完毕

# heartbeat+drbd+mysql的实现

增加master1和master2的resource.d下MySQL脚本文件及修改haresources配置文件：

\cp /data/3306/mysql /etc/ha.d/resource.d/

[root@master1 resource.d]# chmod +x /etc/ha.d/resource.d/mysql

[root@master1 resource.d]# scp -P52113 -rp /etc/ha.d/resource.d/mysql root@172.16.1.52:/etc/ha.d/resource.d/

[root@master1 resource.d]# vim /etc/ha.d/haresources

master1 IPaddr::172.16.1.50/24/eth1 drbddisk::data Filesystem::/dev/drbd0::/data::ext4 mysql

[root@master2 resource.d]# vim /etc/ha.d/haresources

master1 IPaddr::172.16.1.50/24/eth1 drbddisk::data Filesystem::/dev/drbd0::/data::ext4 mysql

重启heartbeat测试服务高可用

主从复制：

从：

[root@slave01 3306]# vim my.cnf

server-id = 4

#long\_query\_time = 1

#log-bin = /data/3306/mysql-bin

主：

[root@master1 3306]# mysql -uroot -poldboy123 -S /data/3306/mysql.sock

mysql> grant replication slave on \*.\* to rep@'172.16.1.%'identified by '123456';

Query OK, 0 rows affected (0.01 sec)

mysql> flush privileges;

Query OK, 0 rows affected (0.01 sec)

[root@master1 3306]# mysqldump -uroot -poldboy123 -S /data/3306/mysql.sock -B -F -R -x --master-data=2 -A --events|gzip >/server/backup/rep3306\_$(date +%F).sql.gz

mysql> show master status;

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

| File | Position | Binlog\_Do\_DB | Binlog\_Ignore\_DB |

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

| mysql-bin.000007 | 329 | | |

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

1 row in set (0.00 sec)

mysql>unlock tables;

把打包的备份文件scp到从库

从：

在外面执行恢复：

gzip -d /server/backup/rep3306\_$(date +%F).sql.gz

mysql -uroot -poldboy123 -S /data/3306/mysql.sock < rep3306\_$(date +%F).sql

进入数据库：

CHANGE MASTER TO

MASTER\_HOST='172.16.1.50',

MASTER\_PORT=3306,

MASTER\_USER='rep',

MASTER\_PASSWORD='123456',

MASTER\_LOG\_FILE='mysql-bin.000007',

MASTER\_LOG\_POS=329;

开启客户端：

start slave;

show slave status\G

最后测试是否同步