**MHA(Master High Availability)**

MySql高可用解决方案，当主宕机后，将主从配置文件中踢出，并从备机中新选出个主

|  |  |
| --- | --- |
| 192.168.4.50 | 客户端 |
| 192.168.4.51 | 主库 |
| 192.168.4.52 | 从库 |
| 192.168.4.53 | 从库 |
| 192.168.4.100 | VIP |
| 192.168.4.57 | 管理主机 |

//准备工作，57对51,52,53面秘钥登录，51,52,53互相免秘钥登录

ssh-keygen -f /root/.ssh/id\_rsa -N ‘’

//51,52,53上安装依赖包,因为mha是用perl语言编写，所以需要perl依赖

yum -y install perl-\* //安装光盘中的依赖

yum -y install mha-soft-student/perl-\* //安装mha的依赖

//配置一主多从，51为主，52,53为从，三台主机都要配置

vim /etc/my.cnf

[mysqld]

log-bin=master51|master 52|master 53

server\_id=51|52|53

//创建主从权限，51主库上创建从库会自动创建

//这个从库上也要创建，因为主库故障后，mha会自动将主库切换到从库上

mysql > grant replication slave on \*.\* to repluser@’%’ identified by ‘123456’;

//创建监控用户

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

//52,53要指定51为主库

mysql >change master to

> master\_host=”192.168.4.51”,

> master\_user=”repluser”,

> master\_password=’123456’,

> master\_log\_file=”master51.000001”,

> master\_log\_pos=154;

mysql > start slave;

//部署MHA集群

cd mha-soft-student/

rpm -ivh mha4mysql-node-0.56-0.el6.noarch.rpm //安装mha-node软件包

tar -zxvf mha4mysql-manager-0.56.tar.gz //解压mha-manager软件包

cd mha4mysql-manager-0.56 //进入源码目录

perl Makefile.PL //配置

make //编译

make install //安装

//配置主配置文件

mkdir /etc/mha //创建工作目录

cp mha4mysql-manager-0.56/sample/conf/app1.cnf /etc/mha/

//拷贝模板

vim /etc/mha/app1.cnf

[server default] //管理服务默认配置

manager\_workdir=/etc/mha //工作目录

manager\_log=/etc/mha/manager.log //日志文件

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

//故障切换脚本

ssh\_user=root //访问ssh服务用户

ssh\_port=22 //ssh服务端口

repl\_user=repluser //主服务器数据同步授权用户

repl\_password=123456 //主从密码

user=root //监控用户

password=123456 //mysql密码

[server1] //指定第1台数据库服务器

hostname=192.168.4.51 //服务器ip地址

port=3306 //服务端口

candidate\_master=1 //竞选主服务器

[server2] //指定第2台数据库服务器

hostname=192.168.4.52

port=3306

candidate\_master=1

[server3] //指定第3台数据库服务器

hostname=192.168.4.53

port=3306

candidate\_master=1

//创建故障切换脚本

cp mha-soft-student/master\_ip\_failover /etc/mha/

vim +35 /etc/mha/master\_ip\_failover

my $vip = '192.168.4.100/24'; # Virtual IP //定义VIP地址

my $key = "1"; //定义变量$key

my $ssh\_start\_vip = "/sbin/ifconfig eth0:$key $vip"; //部署vip地址命令

my $ssh\_stop\_vip = "/sbin/ifconfig eth0:$key down"; //释放vip地址命令

chmod +x /etc/mha/master\_ip\_failover //给脚本加执行权限

//在所有数据库节点上安装mha-node包

cd /root/mha-soft-student/

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

//开启所有节点的半同步模式

vim /etc/my.cnf

[mysqld]

plugin-load="rpl\_semi\_sync\_master=semisync\_master.so;rpl\_semi\_sync\_slave=semisync\_slave.so" //加载模块

rpl\_semi\_sync\_master\_enabled=1 //启用master模块

rpl\_semi\_sync\_slave\_enabled=1 //启用slave模块

relay\_log\_purge=0 //禁止自动删除中继日志文件

//因为从库再主库宕机后会变成主库，所以master和slave模块都要开启

systemctl restatr mysqld

//测试SSH配置,57管理主机上执行

//第一次执行，或者在所有节点上都没有发现vip的情况下需要执行

ifconfig eth0:1 192.168.4.100 //部署VIP

masterha\_check\_ssh --conf=/etc/mha/app1.cnf

.. ..

[info] All SSH connection tests passed successfully.

//测试主从同步

masterha\_check\_repl --conf=/etc/mha/app1.cnf

.. ..

MySQL Replication Health is OK

//启动管理服务

masterha\_manager --conf=/etc/mha/app1.cnf --remove\_dead\_master\_conf \

--ignore\_last\_failover

//remove\_dead\_master\_conf在主库宕机后移除配置文件中的配置

//ignore\_last\_failover在发现主库宕机后mha停止运行，防止再次出现问题

//mha在启动后终端会宕死

//检查mha状态

masterha\_check\_status --conf=/etc/mha/app1.cnf

//停止mha

masterha\_stop --conf=/etc/mha/app1.cnf

//访问集群

mysql -uroot -p123456 -h192.168.4.100

//测试高可用

//停止51的mysql

systemctl stop mysqld

//此时mha监控端会提示51宕机，但4.100还可以ping通，VIP转移到52上,mysql正常使用，主库也转移到52上

ifconfig eth0:1 | grep 4.100

//故障恢复后处理

systemctl start mysqld

//在52上将数据库备份

mysqldump -uroot -p123456 -A --master-data> /root51.sql

scp 51.sql root@192.168.4.51:/root

grep master52 /root/51.sql

CHANGE MASTER TO MASTER\_LOG\_FILE='master52.000001', MASTER\_LOG\_POS=895;

//在51上执行数据恢复,将51变为52的从库

mysql -uroot -p123456 < 51.mysql

mysql > change master to master\_log\_file=’ master52.000001’,master\_log\_pos=’895’;

mysql > start slave;

//mha管理节点57上，重新将51的配置信息加入mha的配置文件中

vim /etc/mha/app1.cnf

[server1 ]

hostname=192.168.4.51

port=3306

candidate\_master=1

//检查ssh，重新启动mha

masterha\_check\_ssh --conf=/etc/mha/app1.cnf

masterha\_manager --conf=/etc/mha/app1.cnf --remove\_dead\_master\_conf \

--ignore\_last\_failover

masterha\_check\_status --conf=/etc/mha/app1.cnf