# 通铭教育实施文档

# 现状需求

因业务需要，增加数据库新环境服务器，为保证数据库的稳定性和高可用性，并且在数据库出现问题时，能够自动切换数据库，选择mysqlMHA高可用架构，并配合maxscale做读写分离。

# 系统架构

MHA（Master High Availability）目前在MySQL高可用方面是一个相对成熟的解决方案，在MySQL故障切换过程中，MHA能做到在0~30秒之内自动完成数据库的故障切换操作，并且在进行故障切换的过程中，MHA能在最大程度上保证数据的一致性，以达到真正意义上的高可用。

服务器信息：

|  |  |
| --- | --- |
| IP | 角色 |
| 10.124.211.80 | master |
| 10.124.211.81 | slave |
| 10.124.211.82 | slave |
| 10.124.211.85 | MHA管理节点 |
| 10.124.211.83 | Maxscale1 |
| 10.124.211.84 | Maxscale2 |
| 10.124.211.221 | Maxcale VIP |

# 数据库实施步骤

## 环境准备

1、创建mysql 用户（三台服务器）

|  |
| --- |
| groupadd -g 27 mysql  useradd -g 27 mysql -G mysql -u 27 mysql |

2、将mysql用户添加到sudo中（所有节点）：

|  |
| --- |
| sudo visudo : |

3、修改mysql家目录:

|  |
| --- |
| sudo /etc/passwd |

## 修改配置参数

三台机器中均修改my.cnf，在最后加入下面参数：

|  |
| --- |
| #复制并行参数  slave\_parallel\_type=LOGICAL\_CLOCK  slave\_parallel\_workers=8  #IO 并行参数  innodb\_read\_io\_threads=8  innodb\_write\_io\_threads=4  #IO动态刷新缓存数据参数  innodb\_io\_capacity=2000  #mysql 临时表大小参数  tmp\_table\_size=16M  #sql排序参数buffer  sort\_buffer\_size=32M  #表连接buffer  join\_buffer\_size=4M  #read buffer  read\_buffer\_size=16M |

## 数据初始化

1、在80、81、82环境中停止数据库服务

|  |
| --- |
| Sudo service mysqld stop |

2、备份81、82上面的数据。

|  |
| --- |
| cd /data  mv mysql mysqll\_bak |

3、将80的数据目录复制到81,82上面

|  |
| --- |
| Sudo scp -r mysql [deployer@10.124.211.81:/data](mailto:deployer@10.162.144.68:/data)  Sudo scp -r mysql [deployer@10.124.211.82:/data](mailto:deployer@10.162.144.69:/data) |

4、启动数据库(三台服务器均执行)

|  |
| --- |
| Sudo service mysqld start |

## 数据库同步配置

分别在81、82上面执行以下命令，配置mysql复制并启动复制：

|  |
| --- |
| CHANGE MASTER TO MASTER\_HOST='10.156.11.131',MASTER\_USER='repl', MASTER\_PASSWORD='\*\*\*\*\*\*',MASTER\_LOG\_FILE='mysql-bin.000001',MASTER\_LOG\_POS=120;  start slave; |

# MHA实施步骤

## 安装mha

### 安装本地yum 源

将cenons iso 包上传服务器，并将其挂载到目录下：

mount -o loop /ztmp/C\* /mnt

修改yum源：

[base]

name=rhel\_dvd

baseurl=file:///mnt/

enabled=1

gpgcheck=0

yum clean all

### 安装相关包

解压后,开始安装相关依赖包：

sudo yum localinstall mysql\*

sudo yum localinstall perl\*

sudo yum localinstall mha\*

## Mha配置

在mha manager节点上，创建masterha相关目录：

cd /etc, mkdir masterha

sudo vi aap1.cnf

|  |
| --- |
| manager\_workdir=/var/log/masterha  manager\_log=/var/log/masterha/app1/manager.log  master\_binlog\_dir=/data/binlog  #master\_ip\_failover\_script=/usr/local/bin/master\_ip\_failover  #master\_ip\_online\_change\_script=/usr/local/bin/master\_ip\_online\_change  password=z5sFf!NjAhdVb3  user=root  ping\_interval=1  remote\_workdir=/tmp  repl\_password=Root#123456  repl\_user=repl  report\_script=/usr/local/bin/send\_report  shutdown\_script=""  ssh\_user=mysql  [server1]  hostname=10.124.211.80  port=3306  candidate\_master=1  check\_repl\_delay=0  [server2]  hostname=10.124.211.81  port=3306  candidate\_master=1  check\_repl\_delay=0  [server3]  hostname=10.124.211.82  port=3306 |

# Maxscale实施步骤

## 环境准备

1、创建mysql 用户（三台服务器）

|  |
| --- |
| groupadd maxscale  useradd -g maxscale -G maxscale maxscale |

2、将mysql用户添加到sudo中（所有节点）：

|  |
| --- |
| sudo visudo : |

3、修改mysql家目录:

|  |
| --- |
| sudo /etc/passwd |

## 安装实施

### 安装依赖

yum install libaio.x86\_64 libaio-devel.x86\_64 novacom-server.x86\_64 libedit -y

MaxScale 的下载地址：

https://downloads.mariadb.com/files/MaxScale

rpm -ivh maxscale-beta-2.0.0-1.centos.7.x86\_64.rpm

在开始配置之前，需要在 master和backup 中为 MaxScale 创建两个用户，用于监控模块和路由模块。

### 创建监控用户

create database maxscale\_schema ;

mysql> create user scalemon@'%' identified by "123456";  
mysql> grant replication slave, replication client on \*.\* to scalemon@'%';

### 创建路由用户

mysql> create user scaleroute@'%' identified by "\*\*\*\*\*\*\*";

mysql> grant select on mysql.\* to scaleroute@'%';

## 配置和启动

### 配置文件

vi /etc/maxscale.cnf：

|  |
| --- |
| [maxscale]  threads=auto  ms\_timestamp=1  syslog=1  maxlog=1  log\_to\_shm=0  log\_warning=1  log\_notice=0  log\_info=0  log\_debug=0  log\_augmentation=1  connection\_timeout=300  max\_connections=5000  logdir=/home/maxscale/log/  datadir=/home/maxscale/data/  cachedir=/home/maxscale/cache/  piddir=/tmp  [server1]  type=server  address=10.124.211.81  port=3306  protocol=MySQLBackend  myweight=5  [server2]  type=server  address=10.124.211.82  port=3306  protocol=MySQLBackend  myweight=5  [server3]  type=server  address=10.124.211.80  port=3306  protocol=MySQLBackend  myweight=5  [MySQL Monitor]  type=monitor  module=mysqlmon  servers=server1,server2,server3  user=maxadmin  passwd=z5sFf)#NjAhdVb3  monitor\_interval=10000  detect\_replication\_lag=true  detect\_stale\_master=true  ##[Read-Only Service]  ###type=service  ###router=readconnroute  ###servers=server1  ###user=root  ###passwd=manager1  ###router\_options=slave  ##  ##  [Read-Write Service]  type=service  router=readwritesplit  servers=server1,server2,server3  user=root  passwd=z5sFf)#NjAhdVb3  weightby=myweight  max\_slave\_connections=100%  enable\_root\_user=1  ##  [MaxAdmin Service]  type=service  router=cli  #  #  [Read-Write Listener]  type=listener  service=Read-Write Service  protocol=MySQLClient  port=3306  ##  [MaxAdmin Listener]  type=listener  service=MaxAdmin Service  protocol=maxscaled  socket=default  # |

### 启动maxscale

|  |
| --- |
| Maxscale -f /etc/maxscale.cnf |

# Keepalived实施步骤

## 安装依赖包

|  |
| --- |
| [deployer@PreOS-dell-26 ztmp]$yum install -y libnl\* libnfnetlink-devel zlib zlib-devel gcc gcc-c++ openssl openssl-devel openssh |

## 安装keepalived

|  |
| --- |
| [deployer@PreOS-dell-26 ~]$ cd /ztmp  [deployer@PreOS-dell-26 ztmp]$ tar -zxvf keepalived-2.0.10.tar.gz  [deployer@PreOS-dell-26 ztmp]$ cd keepalived-2.0.10/  [deployer@PreOS-dell-26 keepalived-2.0.10]$ ./configure --prefix=/usr/local/keepalived  checking for a BSD-compatible install... /bin/install -c  checking whether build environment is sane... yes  checking for a thread-safe mkdir -p... /bin/mkdir -p  checking for gawk... gawk  checking whether make sets $(MAKE)... yes  checking whether make supports nested variables... yes  。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。  。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。  Use IPv4 devconf : No  Use libiptc : No  Use libipset : No  init type : systemd  Strict config checks : No  Build genhash : Yes  Build documentation : No  [deployer@PreOS-dell-26 keepalived-2.0.10]$ make  Making all in lib  make[1]: Entering directory `/ztmp/keepalived-2.0.10/lib'  make all-am  。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。  。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。。  make[1]: Entering directory `/ztmp/keepalived-2.0.10'  EDIT README  make[1]: Leaving directory `/ztmp/keepalived-2.0.10'  [deployer@PreOS-dell-26 keepalived-2.0.10]$ make install  [deployer@PreOS-dell-26 keepalived-2.0.10]$ sudo make install  [sudo] password for deployer:  Making install in lib  make[1]: Entering directory `/ztmp/keepalived-2.0.10/lib'  /bin/mkdir -p '/usr/local/keepalived/share/doc/keepalived'  。。。。。。。。。。。。。。。。。。。。。。。。。。。。  。。。。。。。。。。。。。。。。。。。。。。。。。。。  /bin/install -c -m 644 README '/usr/local/keepalived/share/doc/keepalived'  make[2]: Leaving directory `/ztmp/keepalived-2.0.10'  make[1]: Leaving directory `/ztmp/keepalived-2.0.10'  [deployer@PreOS-dell-26 etc]$ sudo cp /usr/local/keepalived/etc/sysconfig/keepalived /etc/sysconfig/keepalived  [deployer@PreOS-dell-26 etc]$ sudo cp /usr/local/keepalived/sbin/keepalived /usr/sbin/keepalived  [deployer@PreOS-dell-25 init.d]$ sudo scp /ztmp/keepalived-2.0.10/keepalived/etc/init.d/keepalived /etc/init.d/keepalived  [deployer@PreOS-dell-25 init.d]$ sudo chmod +x /etc/init.d/keepalived |

## 配置keepalived

|  |
| --- |
| [deployer@PreOS-dell-26 etc]$ sudo mkdir -p /etc/keepalived  [deployer@PreOS-dell-26 etc]$ sudo vi /etc/keepalived/keepalived.conf  ! Configuration File for keepalived  global\_defs {  ## keepalived  router\_id maxscale01  }  vrrp\_script check\_maxscale {  script " /etc/keepalived/check\_maxscale.sh "  interval 2  }  vrrp\_instance VI\_1 {  state BACKUP  interface bond0.162  virtual\_router\_id 33  priority 101  nopreempt  advert\_int 1  authentication {  auth\_type PASS  auth\_pass KJj23576hYgu23IP  }  virtual\_ipaddress {  10.162.211.221 bond0.162  }  track\_script {  check\_maxscale  }  }  [deployer@PreOS-dell-26 etc]$ sudo vi /etc/keepalvied/check\_maxscale.sh  #!/bin/bash  maxscale\_id=`pidof maxscale`  maxscale\_exist=`ps -ef|grep "maxscale.cnf"|grep -v grep|wc -l`  if [[ $maxscale\_id -eq 0 && $maxscale\_exist -eq 0 ]];then  /etc/init.d/keepalived stop  启动keepalvied  [deployer@PreOS-dell-26 etc]$ /etc/init.d/keepalived start  检查keepalived是否正常运行  [deployer@PreOS-dell-26 etc]$ ps -ef|grep keepalived  root 113428 1 0 11:14 ? 00:00:00 /usr/local/keepalived/sbin/keepalived -D  root 113429 113428 0 11:14 ? 00:00:00 /usr/local/keepalived/sbin/keepalived -D  [deployer@PreOS-dell-26 etc]$ ip a |grep net  inet 127.0.0.1/8 scope host lo  inet6 ::1/128 scope host  inet 160.65.48.125/22 brd 160.65.51.255 scope global bond0  inet6 fe80::f6e9:d4ff:feb6:8d60/64 scope link  inet 10.162.144.84/22 brd 10.162.147.255 scope global bond0.162  **inet 10.162.144.246/32 scope global bond0.162**  inet6 fe80::f6e9:d4ff:feb6:8d60/64 scope link  [deployer@PreOS-dell-26 etc]$   * 1. 验证keepalived切换   在10.162.144.84服务器上执行以下命令:  [deployer@PreOS-dell-26 ~]$ ip a|grep net  inet 127.0.0.1/8 scope host lo  inet6 ::1/128 scope host  inet 160.65.48.125/22 brd 160.65.51.255 scope global bond0  inet6 fe80::f6e9:d4ff:feb6:8d60/64 scope link  inet 10.162.144.84/22 brd 10.162.147.255 scope global bond0.162  **inet 10.162.144.246/32 scope global bond0.162**  inet6 fe80::f6e9:d4ff:feb6:8d60/64 scope link  [deployer@PreOS-dell-26 ~]$ sudo /etc/init.d/keepalived stop  Stopping keepalived (via systemctl): [ OK ]  deployer@PreOS-dell-26 ~]$ ip a |grep net  inet 127.0.0.1/8 scope host lo  inet6 ::1/128 scope host  inet 160.65.48.125/22 brd 160.65.51.255 scope global bond0  inet6 fe80::f6e9:d4ff:feb6:8d60/64 scope link  inet 10.162.144.84/22 brd 10.162.147.255 scope global bond0.162  inet6 fe80::f6e9:d4ff:feb6:8d60/64 scope link  在服务器10.162.144.83上执行以下命令  [deployer@PreOS-dell-25 init.d]$ ip a|grep net  inet 127.0.0.1/8 scope host lo  inet6 ::1/128 scope host  inet 160.65.48.124/22 brd 160.65.51.255 scope global bond0  inet6 fe80::f6e9:d4ff:feb6:7850/64 scope link  inet 10.162.144.83/22 brd 10.162.147.255 scope global bond0.162  **inet 10.162.144.246/32 scope global bond0.162**  inet6 fe80::f6e9:d4ff:feb6:7850/64 scope link  inet 172.17.0.1/16 scope global docker0 |

# 日常维护手册

## 停止集群服务

按照顺序停止MHA，keepalived,maxscale,复制，mysql服务

### 停止MHA

|  |
| --- |
| masterha\_stop --conf=/etc/masterha/app1.cnf  检查mha(管理节点)：  masterha\_check\_status --conf=/etc/masterha/app1.cnf |

### 停止keepalived

|  |
| --- |
| /etc/init.d/keepalived stop  检查keepalived (在83、84服务器上)  ps -ef | grep keepalived 查看是否有进程存在，没有则正常停止 |

### 停止maxscale

|  |
| --- |
| sudo maxadmin -S /home/maxscale/maxscale.sock  >shutdown maxscale |

### 停止复制

|  |
| --- |
| Mysql –uroot –p  Stop slave; |

## 启动集群服务

按照启动顺序启动：启动msql复制，启动MHA，启动maxscale，启动keepalived。

### 启动mysql复制

|  |
| --- |
| Mysql –uroot –p  start slave;  show slave status \G; |

### 启动MHA

|  |
| --- |
| masterha\_stop --conf=/etc/masterha/app1.cnf  检查mha(管理节点:72)：  masterha\_check\_status --conf=/etc/masterha/app1.cnf  masterha\_check\_repl --conf=/etc/masterha/app1.cnf  nohup masterha\_manager --conf=/etc/masterha/app1.cnf --remove\_dead\_master\_conf --ignore\_last\_failover < /dev/null > /var/log/masterha/app1/manager.log 2>&1 &  检查mha(管理节点:72)：  masterha\_check\_status --conf=/etc/masterha/app1.cnf |

### 启动 maxscle

|  |
| --- |
| 在83、84上执行：  Maxscale -f /etc/maxscale.cnf |

### 启动keepalived

|  |
| --- |
| 在83、84上执行：  /etc/init.d/keepalived start |

## 检查MHA状态

|  |
| --- |
| 检查mha(管理节点:72)：  masterha\_check\_status --conf=/etc/masterha/app1.cnf  masterha\_check\_repl --conf=/etc/masterha/app1.cnf |