**使用HAProxy、Galera建立高可用性的Mysql**

我们使用MariaDB( MySQL的分支) 以及Galera

配置参考：

<http://www.360doc.com/content/13/0817/15/834950_307820923.shtml>

# 一、下载并安装MariaDB

1、$apt-get install python-software-properties

2、$sudo apt-key adv --recv-keys --keyserver hkp://keyserver.ubuntu.com:80 0xcbcb082a1bb943db

3、$sudo add-apt-repository 'deb http://ftp.osuosl.org/pub/mariadb/repo/5.5/ubuntu trusty main'

4、$sudo apt-get update

5、$sudo apt-get install mariadb-galera-server galera

6、安装时指定Mysql账户root的密码为RDC.

# 二、编辑MariaDB配置文件

配置第一台mysql节点：mysql1

/etc/mysql/conf.d/cluster.cnf

[mysqld]

query\_cache\_size=0

binlog\_format=ROW

default-storage-engine=innodb

innodb\_file\_per\_table

collation-server = utf8\_general\_ci

init-connect = 'SET NAMES utf8'

character-set-server = utf8

innodb\_autoinc\_lock\_mode=2

query\_cache\_type=0

bind-address=0.0.0.0

# Galera Provider Configuration

wsrep\_provider=/usr/lib/galera/libgalera\_smm.so

#wsrep\_provider\_options="gcache.size=32G"

# Galera Cluster Configuration

wsrep\_cluster\_name="openstack\_mysql\_cluster"

wsrep\_cluster\_address="gcomm://192.168.203.26,192.168.203.27"

# Galera Synchronization Congifuration

wsrep\_sst\_method=rsync

#wsrep\_sst\_auth= wsrep\_sst:wspass

# Galera Node Configuration

wsrep\_node\_address="192.168.203.26"

wsrep\_node\_name="mysql1"

配置第二台节点：mysql2

vi /etc/mysql/conf.d/cluster.cnf

修改以下行：

# Galera Cluster Configuration

wsrep\_cluster\_name="openstack\_mysql\_cluster"

**wsrep\_cluster\_address="gcomm://192.168.203.26"**

（第一台参数的wsrep\_cluster\_address为全部节点，第二台以及以上节点这个参数wsrep\_cluster\_address="gcomm://192.168.203.26"的地址要写成前一台节点的ip地址）

# 三、编辑Mysql配置文件

1、修改配置文件/etc/mysql/my.cnf， 需要将这行注释掉

#bind-address = 127.0.0.1

确定包含以下这行!includedir /etc/mysql/conf.d/

2、增加UTF-8字符集：

[mysqld]

...

default-storage-engine = innodb

innodb\_file\_per\_table

collation-server = utf8\_general\_ci

init-connect = 'SET NAMES utf8'

character-set-server = utf8

# 四、准备启动

1、先停止mysql(两个节点上的mysql都要停止)

$ service mysql stop

# 五、创建集群

1、在第一个节点上启动

$sudo service mysql start --wsrep-new-cluster

2、验证

$mysql -u root -e 'SELECT VARIABLE\_VALUE as "cluster size" FROM INFORMATION\_SCHEMA.GLOBAL\_STATUS WHERE VARIABLE\_NAME="wsrep\_cluster\_size"'

3、在第2个节点上

$sudo service mysql start

4、验证

$ mysql -u root -e 'SELECT VARIABLE\_VALUE as "cluster size" FROM INFORMATION\_SCHEMA.GLOBAL\_STATUS WHERE VARIABLE\_NAME="wsrep\_cluster\_size"'

5、拷贝文件，从第一个节点拷贝到第2个节点

/etc/mysql/debian.cnf

6、重启第2个节点上的mysql

$sudo service mysql stop

$sudo service mysql start

# 六、连接到集群

MariaDB [(none)]> set wsrep\_on=off;

MariaDB [(none)]>set global wsrep\_cluster\_address=’192.168.203.26，192.168.203.27’ （此句可以不执行）

MariaDB [(none)]> show status like 'wsrep%';

# 七、添加HAProxy信息

在任一个节点中，登录mysql

1 、mysql -u root -p

2、grant all on \*.\* to root@'%' identified by 'RDC' with grant option;

3、insert into mysql.user (Host,User) values ('192.168.203.21','haproxy');

4、insert into mysql.user (Host,User) values ('192.168.203.22','haproxy');

5、flush privileges;

6、exit

# 八、在HAProxy节点上安装mysql –client

1、apt-get update

2、apt-get install mysql-client

# 九、修改HAProxy配置文件

编辑 /etc/haproxy/haproxy.cfg

listen galera 192.168.203.20:3306

balance source

mode tcp

option tcpka

option mysql-check user haproxy

server mysql1 192.168.203.26:3306 check weight 1

server mysql2 192.168.203.27:3306 check weight 1

# 十、重启HAProxy

service haproxy reload

# 十一、验证HAProxy对Mysql的负载均衡

在主负载均衡上执行：

mysql -h 192.168.203.20 -u root -p

mysql>

//在上面新建库、表、插入记录，两台mysql集群均插入操作成功！