数据库进阶2

数据读写分离

减轻单台服务器并发访问压力

提高机器硬件的利用率

如何实现数据读写分离

通过程序实现

通过安装软件提供的服务实现（中间件）

mycat, mysql-proxy, maxscale

构建思路

先做主从复制

再做读写分离（需要一台代理器）代理器上需要布置maxscale

修改maxscale配置文件

vim /etc/maxscale.cnf

[maxscale]

threads=auto #开启进程数

[server1] #定义数据库服务器主机名

type=server

address=192.168.4.22 #master主机ip地址

port=3306

protocol=MySQLBackend

[server2] #定义数据库服务器

type=server

address=192.168.4.23 #slave主机ip地址

port=3306

protocol=MySQLBackend

[MySQL Monitor] #定义要监视的数据库节点

type=monitor

module=mysqlmon

servers=server1,server2

user=scalemon #设置监控用户为scalemon，密码为123456（在主从服务器上授权用户）

passwd=123456

monitor\_interval=10000

[Read-Write Service] #定义读写分离的数据库级点

type=service

router=readwritesplit

servers=server1,server2

user=scaleroute #设置路由用户为scaleroute，密码为123456（在主从服务器上授权用户）

passwd=123456

max\_slave\_connections=100%

[MaxAdmin Service]

type=service

router=cli

[Read-Write Listener]

type=listener

service=Read-Write Service

protocol=MySQLClient

port=4006 #设置客户端访问端口为4006

[MaxAdmin Listener]

type=listener

service=MaxAdmin Service

protocol=maxscaled

socket=default

port=4016 #设置本机访问端口为4016

在主从服务器上授权用户

mysql>grant replication slave,replication client on \*.\* to scalemon@”%” identified by “123456”

#创建监控用户（监控客户端的操作）

mysql>grant select on mysql.\* to scaleroute@”%” identified by “1234565”

#创建路由用户（查看mysql库里的表，检测客户端登录的用户是否有授权）

mysql>grant all on \*.\* to yaya@”%” identified by “123456”

#授权yaya用户，客户端登录使用

]#maxscale -f /etc/my.cnf #启用maxscle

]#killall -9 maxscale #关闭maxscale

在代理器上测试

]#maxadmin -uadmin -pmariadb -P4016

在客户端测试

]#mysql -h192.168.4.28 -uyaya -p123456 -P4006

多实例

一台服务器运行多个数据库服务器(提高硬盘利用率，节约运维成本)

安装支持多实例服务的软件包

mysql-5.7.20-linux-glibc2.12-x86\_64.tar.gz

将解压文件移动到/usr/local/mysql下

关掉本机的mysql服务

每个实例拥有独立的数据库目录和监听端口号

要有独立的实例名称和独立的sock文件

修改主配置文件(将原来的配置文件移走)

vim /etc/my.cnf

[mysqld\_multi]

mysqld=/usr/local/mysql/bin/mysqld\_safe

mysqladmin=/usr/local/mysql/bin/mysqladmin

user=root

[mysqld1]

datadir=/dir1

port=3307

log-error=/dir1/mysql3307.log

pid-file=/dir1/mysql3307.pid

socket=/dir1/mysql3307.sock

[mysqld2]

datadir=/dir2

port=3308

log-error=/dir2/mysql3308.log

pid-file=/dir2/mysql3308.pid

socket=/dir2/mysql3308.sock

数据库目录可自行创建

写完配置文件后，启动实例（数据库）

[root@node21 ~]# /usr/local/mysql/bin/mysqld\_multi start 2

[root@node21 ~]# /usr/local/mysql/bin/mysqld\_multi start 1

登录数据库

[root@node21 ~]# /usr/local/mysql/bin/mysql -uroot -p'QA5JheP\_nBpc' -S /dir2/mysql3308.sock

#需指定sock文件

mysql> alter user user() identified by "123456"；

#修改本机登录密码

mysql性能调优

提高mysql系统的性能、响应速度

替换有问题的硬件（cpu/磁盘/内存）

服务程序的运行参数调整

sql查询进行优化