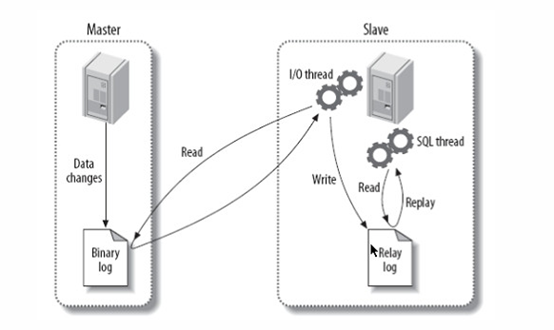
MySQL 半同步的研究和实施

1.    MySQL半同步复制研究背景

半同步主要针对一主多从的生产环境为产生的。MySQL主从复制默认采用的是异步模式，异步模式，从库每次从主库读取binlog日志放到自己的relaylog里，但主库不会关心从库读取多少binlog这样就就不会影响主库的性能，但是当主库宕机，但是从库binglog读取差异很大从而导致数据差异很大。实时同步这样会影响主库的性能。半同步是一个中间产品，他的机制是采用的是实时同步加异步。主库等待从库响应回答，如果超时主库就才用异步同步，当规定时间响应采用同步。系统默认是10s ，mysql5.5 本身自带半同步插件。

2.    半同步逻辑图



3.    环境需求

|  |  |  |
| --- | --- | --- |
| 服务器名 | IP 地址 | MySQL 版本 |
| DB02 | 172.16.1.52 | MySQL5.5.32 |
| M01 | 172.16.1.61 | MySQL5.5.32 |

4. 半同安装

4.1 搭建基础环境

两台主机都安装mysql服务

yum install ncurses-devel libaio-devel -y

rpm -qa ncurses-devel libaio-devel

ncurses-devel-5.7-4.20090207.el6.x86\_64

libaio-devel-0.3.107-10.el6.x86\_64

yum install cmake -y

rpm -qa cmake

useradd mysql -s /sbin/nologin –M

id mysql

cd /home/oldboy/tools

上传mysql安装包到 /home/oldboy/tools

tar xf mysql-5.5.32.tar.gz

cd 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/{3307,3306,3308}/data

[root@m01 mysql-5.5.32]# tree /data

/data

├── 3306

│   └── data

├── 3307

│   └── data

└── 3308

└── data

在3306 3307 3308 分别创建mysql 启动脚本mysql配置文件my.cnf

vim /data/3306/mysql

#!/bin/sh

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

#this scripts is created by oldboy at 2007-06-09

#oldboy QQ:31333741

#site:http://www.etiantian.org

#blog:http://oldboy.blog.51cto.com

#oldboy trainning QQ group: 208160987 226199307  44246017

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

#init

port=3306

mysql\_user="root"

mysql\_pwd="123456"

CmdPath="/application/mysql/bin"

mysql\_sock="/data/${port}/mysql.sock"

#startup function

function\_start\_mysql()

{

    if [ ! -e "$mysql\_sock" ];then

      printf "Starting MySQL...\n"

      /bin/sh ${CmdPath}/mysqld\_safe --defaults-file=/data/${port}/my.cnf 2>&1 > /dev/null &

    else

      printf "MySQL is running...\n"

      exit

    fi

}

#stop function

function\_stop\_mysql()

{

    if [ ! -e "$mysql\_sock" ];then

       printf "MySQL is stopped...\n"

       exit

    else

       printf "Stoping MySQL...\n"

       ${CmdPath}/mysqladmin -u ${mysql\_user} -p${mysql\_pwd} -S /data/${port}/mysql.sock shutdown

   fi

}

#restart function

function\_restart\_mysql()

{

    printf "Restarting MySQL...\n"

    function\_stop\_mysql

    sleep 2

    function\_start\_mysql

}

case $1 in

start)

    function\_start\_mysql

;;

stop)

    function\_stop\_mysql

;;

restart)

    function\_restart\_mysql

;;

\*)

    printf "Usage: /data/${port}/mysql {start|stop|restart}\n"

esac

[root@m01 /]#vim /data/3306/my.cnf

[client]

port            = 3306

socket          = /data/3306/mysql.sock

[mysql]

no-auto-rehash

[mysqld]

user    = mysql

port    = 3306

socket  = /data/3306/mysql.sock

basedir = /application/mysql

datadir = /data/3306/data

open\_files\_limit    = 1024

back\_log = 600

max\_connections = 800

max\_connect\_errors = 3000

table\_cache = 614

external-locking = FALSE

max\_allowed\_packet =8M

sort\_buffer\_size = 1M

join\_buffer\_size = 1M

thread\_cache\_size = 100

thread\_concurrency = 2

query\_cache\_size = 2M

query\_cache\_limit = 1M

query\_cache\_min\_res\_unit = 2k

#default\_table\_type = InnoDB

thread\_stack = 192K

#transaction\_isolation = READ-COMMITTED

tmp\_table\_size = 2M

max\_heap\_table\_size = 2M

long\_query\_time = 1

#log\_long\_format

#log-error = /data/3306/error.log

#log-slow-queries = /data/3306/slow.log

pid-file = /data/3306/mysql.pid

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

relay-log = /data/3306/relay-bin

relay-log-info-file = /data/3306/relay-log.info

binlog\_cache\_size = 1M

max\_binlog\_cache\_size = 1M

max\_binlog\_size = 2M

expire\_logs\_days = 7

key\_buffer\_size = 16M

read\_buffer\_size = 1M

read\_rnd\_buffer\_size = 1M

bulk\_insert\_buffer\_size = 1M

#myisam\_sort\_buffer\_size = 1M

#myisam\_max\_sort\_file\_size = 10G

#myisam\_max\_extra\_sort\_file\_size = 10G

#myisam\_repair\_threads = 1

#myisam\_recover

lower\_case\_table\_names = 1

skip-name-resolve

slave-skip-errors = 1032,1062

replicate-ignore-db=mysql

server-id = 1

innodb\_additional\_mem\_pool\_size = 4M

innodb\_buffer\_pool\_size = 32M

innodb\_data\_file\_path = ibdata1:128M:autoextend

innodb\_file\_io\_threads = 4

innodb\_thread\_concurrency = 8

innodb\_flush\_log\_at\_trx\_commit = 2

innodb\_log\_buffer\_size = 2M

innodb\_log\_file\_size = 4M

innodb\_log\_files\_in\_group = 3

innodb\_max\_dirty\_pages\_pct = 90

innodb\_lock\_wait\_timeout = 120

innodb\_file\_per\_table = 0

[mysqldump]

quick

max\_allowed\_packet = 2M

[mysqld\_safe]

log-error=/data/3306/mysql\_oldboy3306.err

pid-file=/data/3306/mysqld.pid

当配置好了配置文件我们就可以初始化我们的数据库了

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

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

/application/mysql/scripts/mysql\_install\_db --basedir=/application/mysql/ --datadir=/data/3308/data --user=mysql

[root@m01 /]# /data/3306/mysql start

Starting MySQL...

[root@m01 /]# /data/3307/mysql start

Starting MySQL...

[root@m01 /]# /data/3308/mysql start

Starting MySQL...

[root@m01 /]# ss -lntup|grep 330

tcp    LISTEN     0      600                    \*:3306                  \*:\*      users:(("mysqld",80959,12))

tcp    LISTEN     0      600                    \*:3307                  \*:\*      users:(("mysqld",81677,11))

tcp    LISTEN     0      600                    \*:3308                  \*:\*      users:(("mysqld",82419,12))

设置root 密码

[root@m01 /]# mysqladmin  -uroot password 123456 -S /data/3306/mysql.sock

[root@m01 /]# mysqladmin  -uroot password 123456 -S /data/3307/mysql.sock

[root@m01 /]# mysqladmin  -uroot password 123456 -S /data/3308/mysql.sock

4.2 搭建主从复制

基础都搭建好了，我们需要设置mysql主从同步

主从复制首先要确定主库开启log-bin

[root@m01 /]# egrep "log-bin" /data/3306/my.cnf

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

授权rep用户用于复制

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

Query OK, 0 rows affected (0.01 sec)

mysql> show grants for  rep@"172.16.1.%" ;

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

| Grants for rep@172.16.1.%                                                                                               |

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

| GRANT REPLICATION SLAVE ON \*.\* TO 'rep'@'172.16.1.%' IDENTIFIED BY PASSWORD '\*6BB4837EB74329105EE4568DDA7DC67ED2CA2AD9' |

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

1 row in set (0.00 sec)

备份数据库并导入到从库里

 [root@m01 /]# mysqldump -uroot -p123456 -S /data/3306/mysql.sock  --master-data=2 -A -B --events -x |gzip >/server/backup/all\_bak\_$(date +%F).sql.gz

[root@m01 /]# ls /server/backup/

all\_bak\_2015-12-19.sql.gz

远程拷贝到对方主机上

[root@m01 /]# scp -P52113 -r /server/backup/all\_bak\_2015-12-19.sql.gz  [root@172.16.1.52:/server/backup/](mailto:root@172.16.1.52:/server/backup/)

在db02上导入数据库看是否成功

mysql  -uroot -p123456 -S /data/3308/mysql.sock <all\_bak\_2015-12-19.sql

CHANGE MASTER TO   MASTER\_HOST='172.16.1.61',   MASTER\_PORT=3306,  MASTER\_USER='rep',   MASTER\_PASSWORD='123456';

CHANGE MASTER TO MASTER\_LOG\_FILE='mysql-bin.000001', MASTER\_LOG\_POS=474;

[root@db02 backup]#  mysql -uroot -p123456 -S /data/3306/mysql.sock  -e " show slave status \G;"|egrep  -i  "\_running|\_behind"

             Slave\_IO\_Running: Yes

            Slave\_SQL\_Running: Yes

        Seconds\_Behind\_Master: 0

当从库设置成功了

4.3 安装mysql 半同步插件

安装mysql半同插件， mysql5.5 已经自带这个插件

1.    查看半同步插件的位置

mysql> show variables like 'plugin\_dir';

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

| Variable\_name | Value                          |

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

| plugin\_dir    | /application/mysql/lib/plugin/ |

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

1 row in set (0.00 sec)

[root@m01 /]# ll  /application/mysql/lib/plugin/

total 876

-rwxr-xr-x 1 root root  13038 Dec 19 19:06 adt\_null.so

-rwxr-xr-x 1 root root  24683 Dec 19 19:06 auth.so

-rwxr-xr-x 1 root root  12564 Dec 19 19:06 auth\_socket.so

-rwxr-xr-x 1 root root  22977 Dec 19 19:06 auth\_test\_plugin.so

-rw-r--r-- 1 root root    227 Jul  2  2013 daemon\_example.ini

drwxr-xr-x 2 root root   4096 Dec 19 19:24 debug

-rwxr-xr-x 1 root root 421101 Dec 19 19:02 ha\_archive.so

-rwxr-xr-x 1 root root  28380 Dec 19 19:06 libdaemon\_example.so

-rwxr-xr-x 1 root root  17779 Dec 19 19:06 mypluglib.so

-rwxr-xr-x 1 root root  17567 Dec 19 19:06 qa\_auth\_client.so

-rwxr-xr-x 1 root root  23278 Dec 19 19:06 qa\_auth\_interface.so

-rwxr-xr-x 1 root root  12902 Dec 19 19:06 qa\_auth\_server.so

-rwxr-xr-x 1 root root 172984 Dec 19 19:06 semisync\_master.so

-rwxr-xr-x 1 root root  94098 Dec 19 19:06 semisync\_slave.so

在master安装

mysql>  install plugin rpl\_semi\_sync\_master soname 'semisync\_master.so';

在slave 安装

mysql>  install plugin rpl\_semi\_sync\_slave soname 'semisync\_slave.so';

安装完毕检查

mysql> show plugins;

查看当前半同步状态

mysql> SHOW VARIABLES LIKE 'rpl\_semi\_sync%';

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

| Variable\_name                      | Value |

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

| rpl\_semi\_sync\_master\_enabled       | OFF   |

| rpl\_semi\_sync\_master\_timeout       | 10000 |

| rpl\_semi\_sync\_master\_trace\_level   | 32    |

| rpl\_semi\_sync\_master\_wait\_no\_slave | ON    |

mysql> SHOW VARIABLES LIKE 'rpl\_semi\_sync%';

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

| Variable\_name                   | Value |

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

| rpl\_semi\_sync\_slave\_enabled     | OFF   |

| rpl\_semi\_sync\_slave\_trace\_level | 32    |

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

开启半同开关

master 端开启

mysql> SET GLOBAL rpl\_semi\_sync\_master\_enabled = 1;

mysql> SHOW VARIABLES LIKE 'rpl\_semi\_sync%';

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

| Variable\_name                      | Value |

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

| rpl\_semi\_sync\_master\_enabled       | ON    |

| rpl\_semi\_sync\_master\_timeout       | 10000 |

| rpl\_semi\_sync\_master\_trace\_level   | 32    |

| rpl\_semi\_sync\_master\_wait\_no\_slave | ON    |

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

slave 开启

mysql> SET GLOBAL rpl\_semi\_sync\_slave\_enabled = 1;

Query OK, 0 rows affected (0.00 sec)

mysql> SHOW VARIABLES LIKE 'rpl\_semi\_sync%';

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

| Variable\_name                   | Value |

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

| rpl\_semi\_sync\_slave\_enabled     | ON    |

| rpl\_semi\_sync\_slave\_trace\_level | 32    |

slave 上需要重启slave

mysql> stop slave IO\_THREAD;

Query OK, 0 rows affected (0.00 sec)

mysql> start slave IO\_THREAD;

Query OK, 0 rows affected (0.00 sec)

master 查看半同步状态

mysql> show status like 'Rpl\_semi\_sync%';

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

| Variable\_name                              | Value |

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

| Rpl\_semi\_sync\_master\_clients               | 1     |

| Rpl\_semi\_sync\_master\_net\_avg\_wait\_time     | 0     |

| Rpl\_semi\_sync\_master\_net\_wait\_time         | 0     |

| Rpl\_semi\_sync\_master\_net\_waits             | 0     |

| Rpl\_semi\_sync\_master\_no\_times              | 0     |

| Rpl\_semi\_sync\_master\_no\_tx                 | 0     |

| Rpl\_semi\_sync\_master\_status                | ON    |

| Rpl\_semi\_sync\_master\_timefunc\_failures     | 0     |

| Rpl\_semi\_sync\_master\_tx\_avg\_wait\_time      | 0     |

| Rpl\_semi\_sync\_master\_tx\_wait\_time          | 0     |

| Rpl\_semi\_sync\_master\_tx\_waits              | 0     |

| Rpl\_semi\_sync\_master\_wait\_pos\_backtraverse | 0     |

| Rpl\_semi\_sync\_master\_wait\_sessions         | 0     |

| Rpl\_semi\_sync\_master\_yes\_tx                | 0     |

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

14 rows in set (0.00 sec)

| Rpl\_semi\_sync\_master\_clients              | 0     |   #记录支持半同步的slave的个数

| Rpl\_semi\_sync\_master\_net\_avg\_wait\_time    | 0     |   #master 等待slave回复的平均等待时间，单位微秒

| Rpl\_semi\_sync\_master\_net\_wait\_time        | 0     |   #master 总的等待时间

| Rpl\_semi\_sync\_master\_net\_waits           | 0   |   #master 等待slave回复的的总的等待次数

| Rpl\_semi\_sync\_master\_no\_times           | 0     |   #master 关闭半同步复制的次数

| Rpl\_semi\_sync\_master\_no\_tx              | 0     |   #master 没有收到slave的回复而提交的次数

| Rpl\_semi\_sync\_master\_status                | OFF   |   #标记master现在是否是半同步复制状态

| Rpl\_semi\_sync\_master\_timefunc\_failures     | 0     |   #时间函数未正常工作的次数

| Rpl\_semi\_sync\_master\_tx\_avg\_wait\_time     | 0     |   #开启Semi-sync，事务返回需要等待的平均时间

| Rpl\_semi\_sync\_master\_tx\_wait\_time          | 0     |   #事务等待备库响应的总时间

| Rpl\_semi\_sync\_master\_tx\_waits              | 0     |   #事务等待备库响应的总次数

| Rpl\_semi\_sync\_master\_wait\_pos\_backtraverse  | 0     |   #改变当前等待最小二进制日志的次数

| Rpl\_semi\_sync\_master\_wait\_sessions         | 0     |   #当前有多少个session 因为slave 的回复而造成等待

| Rpl\_semi\_sync\_master\_yes\_tx                | 0     |   #master 成功接收到slave的回复的次数

5.半同步测试

mysql> insert into t1 values (1);

Query OK, 1 row affected (0.01 sec)

mysql> insert into t1 values (2);

Query OK, 1 row affected (0.01 sec)

mysql> insert into t1 values (3);

Query OK, 1 row affected (0.00 sec)

mysql> show status like 'Rpl\_semi\_sync%';

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

| Variable\_name                              | Value |

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

| Rpl\_semi\_sync\_master\_clients               | 1     |

| Rpl\_semi\_sync\_master\_net\_avg\_wait\_time     | 3192  |

| Rpl\_semi\_sync\_master\_net\_wait\_time         | 15963 |

| Rpl\_semi\_sync\_master\_net\_waits             | 5     |

| Rpl\_semi\_sync\_master\_no\_times              | 0     |

| Rpl\_semi\_sync\_master\_no\_tx                 | 0     |

| Rpl\_semi\_sync\_master\_status                | ON    |

| Rpl\_semi\_sync\_master\_timefunc\_failures     | 0     |

| Rpl\_semi\_sync\_master\_tx\_avg\_wait\_time      | 3121  |

| Rpl\_semi\_sync\_master\_tx\_wait\_time          | 15605 |

| Rpl\_semi\_sync\_master\_tx\_waits              | 5     |

| Rpl\_semi\_sync\_master\_wait\_pos\_backtraverse | 0     |

| Rpl\_semi\_sync\_master\_wait\_sessions         | 0     |

| Rpl\_semi\_sync\_master\_yes\_tx                | 5     |

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

14 rows in set (0.00 sec)

从库关闭 slave IO

mysql> stop slave IO\_THREAD;

再在主库插入数据再看同步状态

mysql> insert into t1 values (1);

Query OK, 1 row affected (3.00 sec)

mysql> show status like 'Rpl\_semi\_sync%';

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

| Variable\_name                              | Value |

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

| Rpl\_semi\_sync\_master\_clients               | 1     |

| Rpl\_semi\_sync\_master\_net\_avg\_wait\_time     | 3597  |

| Rpl\_semi\_sync\_master\_net\_wait\_time         | 46766 |

| Rpl\_semi\_sync\_master\_net\_waits             | 13    |

| Rpl\_semi\_sync\_master\_no\_times              | 2     |

| Rpl\_semi\_sync\_master\_no\_tx                 | 2     |

| Rpl\_semi\_sync\_master\_status                | OFF   |

| Rpl\_semi\_sync\_master\_timefunc\_failures     | 0     |

| Rpl\_semi\_sync\_master\_tx\_avg\_wait\_time      | 2724  |

| Rpl\_semi\_sync\_master\_tx\_wait\_time          | 27244 |

| Rpl\_semi\_sync\_master\_tx\_waits              | 10    |

| Rpl\_semi\_sync\_master\_wait\_pos\_backtraverse | 0     |

| Rpl\_semi\_sync\_master\_wait\_sessions         | 0     |

| Rpl\_semi\_sync\_master\_yes\_tx                | 10    |

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

当在slave上开启

mysql> show status like 'Rpl\_semi\_sync%';

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

| Variable\_name                              | Value |

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

| Rpl\_semi\_sync\_master\_clients               | 1     |

| Rpl\_semi\_sync\_master\_net\_avg\_wait\_time     | 4904  |

| Rpl\_semi\_sync\_master\_net\_wait\_time         | 73571 |

| Rpl\_semi\_sync\_master\_net\_waits             | 15    |

| Rpl\_semi\_sync\_master\_no\_times              | 2     |

| Rpl\_semi\_sync\_master\_no\_tx                 | 3     |

| Rpl\_semi\_sync\_master\_status                | ON    |

| Rpl\_semi\_sync\_master\_timefunc\_failures     | 0     |

| Rpl\_semi\_sync\_master\_tx\_avg\_wait\_time      | 3189  |

| Rpl\_semi\_sync\_master\_tx\_wait\_time          | 35086 |

| Rpl\_semi\_sync\_master\_tx\_waits              | 11    |

| Rpl\_semi\_sync\_master\_wait\_pos\_backtraverse | 0     |

| Rpl\_semi\_sync\_master\_wait\_sessions         | 0     |

| Rpl\_semi\_sync\_master\_yes\_tx                | 11    |