XtraBackup备份工具学习笔记

马飞

2017-12-5-6

# 一、环境概述

xtrabackup是Percona公司CTO Vadim参与开发的一款基于InnoDB的在线热备工具，具有开源，免费，支持在线热备，备份恢复速度快，占用磁盘空间小等特点，并且支持不同情况下的多种备份形式。

xtrabackup的官方下载地址为http://www.percona.com/software/percona-xtrabackup。

xtrabackup包含两个主要的工具，即xtrabackup和innobackupex，二者区别如下：

（1）xtrabackup只能备份innodb和xtradb两种引擎的表，而不能备份myisam引擎的表；

（2）innobackupex是一个封装了xtrabackup的Perl脚本，支持同时备份innodb和myisam，但在对myisam备份时需要加一个全局的读锁。还有就是myisam不支持增量备份。

**测试环境：**

**操作系统：**CentOS6.7

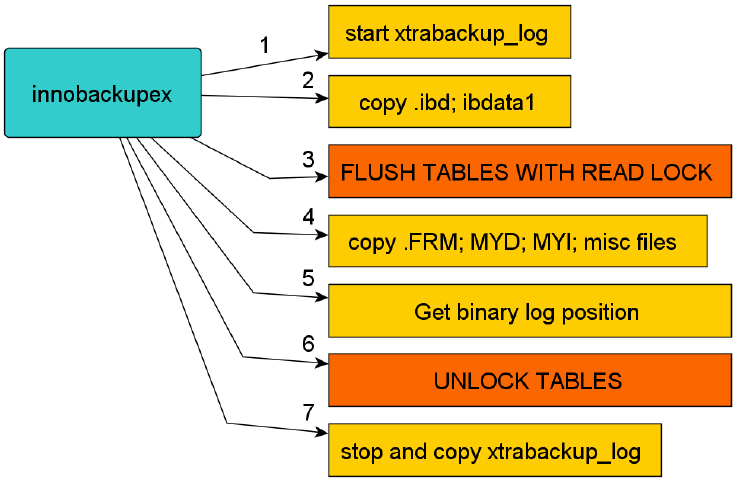
数据库版本：MySQL5.6.31

MySQL5.6 安装采用二进制手动安装方式

<http://www.zhitbar.com/59.html>

# 二、备份过程

innobackupex备份过程如下图：

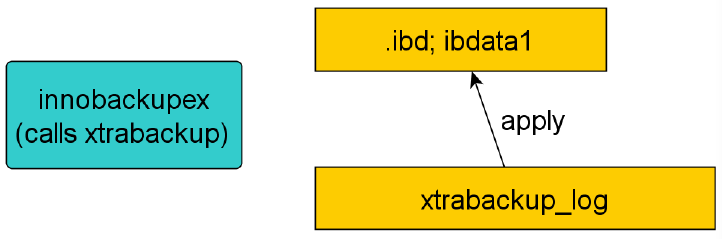


（图1 innobackupex备份过程,本文中所有图都是google所得）

在图1中，备份开始时首先会开启一个后台检测进程，实时检测mysql redo的变化，一旦发现redo中有新的日志写入，立刻将日志记入后台日志文件xtrabackup\_log中。之后复制innodb的数据文件和系统表空间文件ibdata1，待复制结束后，执行flush tables with read lock操作，复制.frm，MYI，MYD，等文件（执行flush tableswith read lock的目的是为了防止数据表发生DDL操作，并且在这一时刻获得binlog的位置）最后会发出unlock tables，把表设置为可读可写状态，最终停止xtrabackup\_log。

# 三、全备恢复

这一阶段会启动xtrabackup内嵌的innodb实例，回放xtrabackup日志xtrabackup\_log，将提交的事务信息变更应用到innodb数据/表空间，同时回滚未提交的事务(这一过程类似innodb的实例恢复）。恢复过程如下图：

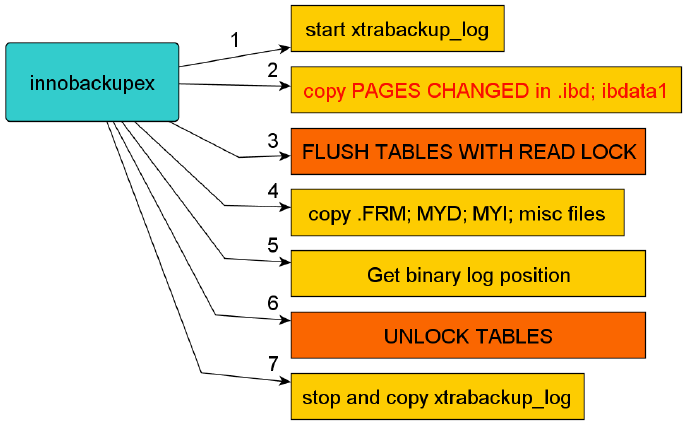


（图2 innobackupex 恢复过程）

# 四、增量备份

innobackupex增量备份过程中的"增量"处理，其实主要是相对innodb而言，对myisam和其他存储引擎而言，它仍然是全拷贝(全备份)

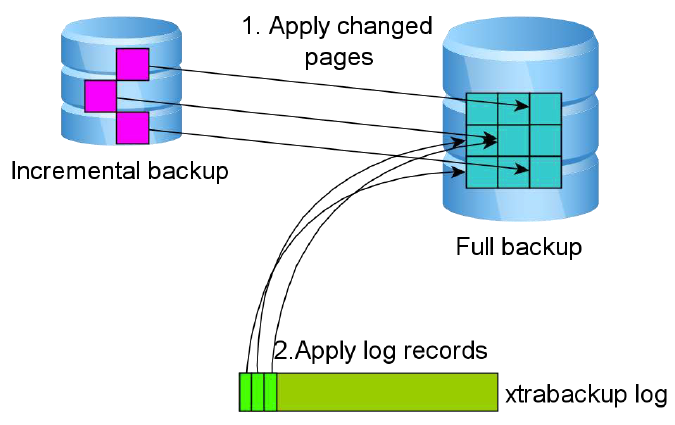
"增量"备份的过程主要是通过拷贝innodb中有变更的"页"（这些变更的数据页指的是"页"的LSN大于xtrabackup\_checkpoints中给定的LSN）。增量备份是基于全备的，第一次增备的数据必须要基于上一次的全备，之后的每次增备都是基于上一次的增备，最终达到一致性的增备。增量备份的过程如下，和全备的过程很类似，区别仅在第2步。



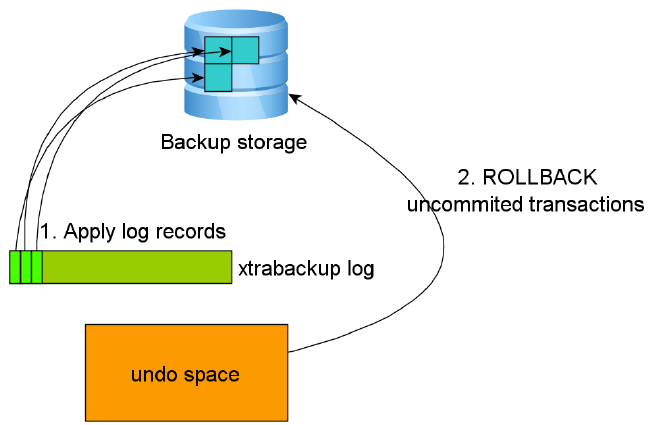
（ 图 3 innobackupex增量备份过程）

# 五、增量备份恢复

和全备恢复类似，也需要两步，一是数据文件的恢复，如图4，这里的数据来源由3部分组成：全备份，增量备份和xtrabackup log。二是对未提交事务的回滚，如图5所示：



（ 图4 innobackupex 增量备份恢复过程1）



( 图5 innobackupex增量备份恢复过程2）

# 五、innobackupex使用示例

## 5.1 安装xtrabackup

安装使用xtrabackup，安装比较简单，我们使用二进制编译好的就行了。

|  |
| --- |
| xtrabackup 2.4.8:  wget <https://www.percona.com/downloads/XtraBackup/Percona-XtraBackup-2.4.8/binary/tarball/percona-xtrabackup-2.4.8-Linux-x86_64.tar.gz> |

|  |
| --- |
| [root@db1] # tar zxvf percona-xtrabackup-2.4.8.tar.gz -C /usr/local/  [root@db1] # mv /usr/local/percona-xtrabackup-2.4.8 /usr/local/xtrabackup  [root@db1] # echo "export PATH=\$PATH:/usr/local/xtrabackup/bin" >> /etc/profile  [root@db1] # source /etc/profile |

## 5.2 全量备份

### 5.2.1创建备份用户

|  |
| --- |
| mysql> create user 'backup'@'%' identified by 'mafei';  mysql> grant process,reload,lock tables,replication client,create tablespace,super on \*.\* to 'backup'@'%';  mysql>flush privileges; |

### 5.2.2 进行全备份

备份数据存放在/data/backup/下面，innobackupex会自动创建一个文件夹，是当前系统的时间戳, 测试数据就是test库中的xs表。

|  |
| --- |
| mysql> create database test;  mysql> use test;  mysql> create table xs(xh int,xm varchar(20),xb int,nl int,csrq date);  insert into xs(xh,xm,xb,nl,csrq) values(1,'zhang.san',1,20,'2009-1-20');  insert into xs(xh,xm,xb,nl,csrq) values(2,'li.shi',0,30,'1996-1-20'); |

|  |
| --- |
| [root@MySQL-01 ~]# innobackupex --defaults-file=/etc/my.cnf --user=backup --password=mafei --socket=/tmp/mysql.sock /data/backup/    innobackupex --defaults-file=/etc/my.cnf --user=backup --password=mafei --socket=/opt/mysql/data/mysql.sock /opt/db\_backup/mysql\_bak/xtrabackup |

**--lock-ddl-per-table**

Lock DDL for each table before xtrabackup starts to copy it and until the backup is completed.

**--kill-long-queries-timeout=60**

This option specifies the number of seconds innobackupex waits between starting FLUSH TABLES WITH READ LOCK and killing those queries that block it. Default is 0

seconds, which means innobackupex will not attempt to kill any queries.

|  |
| --- |
| innobackupex --defaults-file=/etc/my.cnf --user=backup --password=mafei --lock-ddl-per-table /opt/db\_backup/mysql\_bak/xtrabackup/dump |

上面的过程中处理过，主要看最后是否提示innobackupex completed ok，可以看见备份成功。我们看看/data/backup目录下产生了什么

|  |
| --- |
| [root@db1 backup]# pwd  /data/backup  [root@db1 backup]# ll  总用量 4  drwxr-x---. 12 root root 4096 12月 1 23:54 2017-12-01\_23-54-13  [root@db1 backup]# cd 2017-12-01\_23-54-13/  [root@db1 2017-12-02\_06-08-33]# ll  总用量 12316  -rw-r-----. 1 root root 418 12月 2 06:08 backup-my.cnf  -rw-r-----. 1 root root 12582912 12月 2 06:08 ibdata1  drwxr-x---. 2 root root 4096 12月 2 06:08 mysql  drwxr-x---. 2 root root 4096 12月 2 06:08 performance\_schema  drwxr-x---. 2 root root 4096 12月 2 06:08 test  -rw-r-----. 1 root root 113 12月 2 06:08 xtrabackup\_checkpoints  -rw-r-----. 1 root root 487 12月 2 06:08 xtrabackup\_info  -rw-r-----. 1 root root 2560 12月 2 06:08 xtrabackup\_logfile |

可以看见有对应数据库的名字，比如test，还有一个以时间戳命名的目录。我们看看对应文件里面的内容，这几个比较重要

|  |
| --- |
| [root@db1 2017-12-01\_23-54-13]# cat xtrabackup\_checkpoints  backup\_type = full-backuped  from\_lsn = 0  to\_lsn = 3446733  last\_lsn = 3446733  compact = 0  recover\_binlog\_info = 0  [root@db1 2017-12-01\_23-54-13]# more xtrabackup\_info  uuid = e22bef77-d6af-11e7-9f0c-000c29b9baa6  name =  tool\_name = innobackupex  tool\_command = --defaults-file=/etc/my.cnf --user=backup --password=... --socket=/var/lib/mysql/mysql.sock /data/backup/  tool\_version = 2.4.8  ibbackup\_version = 2.4.8  server\_version = 5.6.36  start\_time = 2017-12-01 23:54:13  end\_time = 2017-12-01 23:54:16  lock\_time = 0  binlog\_pos =  innodb\_from\_lsn = 0  innodb\_to\_lsn = 3446733  partial = N  incremental = N  format = file  compact = N  compressed = N  encrypted = N |

可以看见相关文件记录了LSN,日志偏移量，还可以看见这次是全备份。

### 5.2.3 删除数据库

|  |
| --- |
| mysql> drop database test;  Query OK, 1 row affected (0.04 sec)  mysql> |

### 5.2.4 恢复全备

恢复备份到mysql的数据文件目录，这一过程要先关闭mysql数据库，重命名或者删除原数据文件目录都可以，再创建一个新的数据文件目录，将备份数据复制到新的数据文件目录下，赋权，修改权限，启动数据库。

|  |
| --- |
| [root@db1 lib]# /etc/init.d/mysqld stop  Shutting down MySQL..... [ OK ]  [root@db1 lib]# cd /data  [root@db1 lib]# mv mysql mysql\_bak  [root@db1 lib]# mkdir –p /data/mysql/data  [root@db1 lib]# chown -R mysql:mysql /data/mysql/data |

|  |
| --- |
| [root@db1 ~]# innobackupex --apply-log /data/backup/2017-12-01\_23-54-13/ |

以上对应的目录就是innobackupex全备份自己创建的目录。

|  |
| --- |
| [root@db1 mysql]# innobackupex --defaults-file=/etc/my.cnf --copy-back --rsync /data/backup/2017-12-01\_23-54-13/ |

通过查询日志，可以看见已经成功恢复，修改数据目录权限，启动mysql，效验数据是否正常，查看yayun库下面的t1表中的数据。

|  |
| --- |
| [root@MySQL-01 ~]# chown -R mysql.mysql /data/mysql/data  [root@MySQL-01 ~]# service mysql start  Starting MySQL................. [ OK ]  [root@MySQL-01 ~]# |

|  |
| --- |
| mysql> use test;  Reading table information for completion of table and column names  You can turn off this feature to get a quicker startup with -A  Database changed  mysql> select \* from xs;  +----+------------------+-------+----------+-----------------+  | xh | xm | xb | nl | csrq |  +----+------------------+--------+---------+------------------+  | 1 | zhang.san | 1 | 20 | 2009-01-20 |  | 2 | li.shi | -0 | 30 | 1996-01-20 |  +------+----------------+--------+---------+------------------+  2 rows in set (0.00 sec) |

发现数据已经成功恢复。

## 5.3 增量备份

在进行增量备份时，首先要进行一次全量备份，第一次增量备份是基于全备的，之后的增量备份是基于上一次的增量备份，以此类推。

全备份放在/data/backup/full,增量备份放在/data/backup/incremental

[root@db1 backup]# mkdir /data/backup/full

[root@db1 backup]# mkdir /data/backup/incremental

[root@MySQL-01 ~]# tree /data/backup/

/data/backup/

├── full

└── incremental

2 directories, 0 files

### 5.3.1 进行全量备份

|  |
| --- |
| innobackupex --defaults-file=/etc/my.cnf --user=backup --password=mafei --socket=/tmp/mysql.sock /data/backup/full/  [root@db1 incremental]# ls -l /data/backup/full  总用量 4  drwxr-x---. 5 root root 4096 12月 2 08:19 2017-12-02\_08-12-00 |

### 5.3.2 增量备份集一

我们向xs表插入数据，然后创建增量备份1

|  |
| --- |
| use test  insert into xs(xh,xm,xb,nl,csrq) values(3,'wang.wu',1,34,'2002-5-7');  insert into xs(xh,xm,xb,nl,csrq) values(4,'zhao.liu',1,42,'1975-1-9'); |

|  |
| --- |
| innobackupex --defaults-file=/etc/my.cnf --user=backup --password=mafei --socket=/tmp/mysql.sock --incremental /data/backup/incremental/ --incremental-basedir=/data/backup/full/2017-12-02\_08-12-00 --parallel=2  [root@db1 incremental]# ls -l /data/backup/incremental/  总用量 8  drwxr-x---. 5 root root 4096 12月 2 08:17 2017-12-02\_08-12-47 |

增量备份的大小以及文件内容：

|  |
| --- |
| [root@db1 incremental]# du -sh /data/backup/full/2017-12-02\_08-12-00/  131M /data/backup/full/2017-12-02\_08-12-00/  [root@db1 incremental]# du -sh /data/backup/incremental/2017-12-02\_08-12-47/  11M /data/backup/incremental/2017-12-02\_08-12-47/ |

看见增量备份的数据很小吧，就是备份改变的数据而已。

|  |
| --- |
| cd /data/backup/incremental/2017-12-02\_08-12-47/  [root@db1 2017-12-02\_08-12-47]# more xtrabackup\_checkpoints  backup\_type = incremental  from\_lsn = 1649537  to\_lsn = 1651228  last\_lsn = 1651228  compact = 0  recover\_binlog\_info = 0 |

上面已经明显说明是增量备份了，该工具很人性化吧，呵呵

### 5.3.4 增量备份集二

我们再次向xs表插入数据，然后创建增量备份2

|  |
| --- |
| insert into xs(xh,xm,xb,nl,csrq) values(5,'wang.wu2',0,34,'2002-5-7');  insert into xs(xh,xm,xb,nl,csrq) values(6,'zhao.liu2',0,42,'1975-1-9'); |

注意下面标红部分：

|  |
| --- |
| innobackupex --defaults-file=/etc/my.cnf --user=backup --password=mafei --socket=/tmp/mysql.sock --incremental /data/backup/incremental/ --incremental-basedir=/data/backup/incremental/2017-12-02\_08-12-47/ --parallel=2  [root@db1 incremental]# ls -l /data/backup/incremental/  总用量 8  drwxr-x---. 5 root root 4096 12月 2 08:17 2017-12-02\_08-12-47  drwxr-x---. 5 root root 4096 12月 2 08:18 2017-12-02\_08-14-29 |

增量备份的大小以及文件内容：

|  |
| --- |
| [root@db1 incremental]# du -sh /data/backup/incremental/2017-12-02\_08-14-29/  11M /data/backup/incremental/2017-12-02\_08-14-29/ |

### 5.3.5 增量备份恢复

增量备份的恢复大体为3个步骤：

\*恢复完全备份

\*恢复增量备份到完全备份（开始恢复的增量备份要添加--redo-only参数，到最后一次增量备份去掉--redo-only参数）

\*对整体的完全备份进行恢复，回滚那些未提交的数据

（1）恢复备份到mysql的数据文件目录，这一过程要先关闭mysql数据库，重命名或者删除原数据文件目录都可以，再创建一个新的数据文件目录，将备份数据复制到新的数据文件目录下，赋权，修改权限，启动数据库。

|  |
| --- |
| [root@db1 mysql]# service mysql stop  Shutting down MySQL.. SUCCESS!  [root@db1 lib]# cd /data  [root@db1 lib]# mv mysql mysql\_bak  [root@db1 lib]# mkdir -p /data/mysql/data  [root@db1 lib]# chown -R mysql:mysql /data/mysql/data |

（2）恢复完全备份（注意这里一定要加--redo-only参数，该参数的意思是只应用xtrabackup日志中已提交的事务数据，不回滚还未提交的数据）。

|  |
| --- |
| [root@db1 ~]# innobackupex --apply-log --redo-only /data/backup/full/2017-12-02\_08-12-00/ |

（3）将增量备份1应用到完全备份

|  |
| --- |
| innobackupex --apply-log --redo-only /data/backup/full/2017-12-02\_08-12-00/ --incremental-dir=/data/backup/incremental/2017-12-02\_08-12-47/ |

（4）将增量备份2应用到完全备份

|  |
| --- |
| [root@db1 backup]# innobackupex --apply-log /data/backup/full/2017-12-02\_08-12-00/ --incremental-dir=/data/backup/incremental/2017-12-02\_08-14-29/ |

注意：恢复最后一个增量备份时需要去掉--redo-only参数，回滚xtrabackup日志中那些还未提交的数据。

（5）把所有合在一起的完全备份整体进行一次apply操作，回滚未提交的数据。

|  |
| --- |
| innobackupex --apply-log /data/backup/full/2017-12-02\_08-12-00/ |

（6）把恢复完的备份复制到数据库目录文件中，赋权，然后启动mysql数据库，检测数据正确性。

|  |
| --- |
| innobackupex --defaults-file=/etc/my.cnf --copy-back --rsync /data/backup/full/2017-12-02\_08-12-00/  chown -R mysql:mysql /data/mysql/data/  [root@db1 data]# service mysql start  Starting MySQL. SUCCESS!  [root@db1 data]# mysql -uroot -p123456 test  mysql> select \* from xs;  +---------+----------------+----------+--------+-----------------+  | xh | xm | xb | nl | csrq |  +--------+------------------+---------+--------+-----------------+  | 1 | zhang.san | 1 | 20 | 2009-01-20 |  | 2 | li.shi | 0 | 30 | 1996-01-20 |  | 3 | wang.wu | 1 | 34 | 2002-05-07 |  | 4 | zhao.liu | 1 | 42 | 1975-01-09 |  | 5 | wang.wu2 | 0 | 34 | 2002-05-07 |  | 6 | zhao.liu2 | 0 | 42 | 1975-01-09 |  +--------+------------------+---------+--------+-----------------+  6 rows in set (0.00 sec) |

至此,数据恢成功!

# 六、参考

<https://www.cnblogs.com/gomysql/p/3650645.html>

# 七、错误

版本：percona-xtrabackup-2.4.12-Linux-x86\_64.libgcrypt20.tar.gz

错误：innobackupex --defaults-file=/etc/my.cnf --user=backup --password=mafei --socket=/opt/mysql/data/mysql.sock /opt/db\_backup/mysql\_bak/xtrabackup

innobackupex: error while loading shared libraries: libgcrypt.so.20: cannot open shared object file: No such file or directory

解决：yum install libgcrypt –y

yum install numactl –y

yum install perl-DBI -y

yum install perl-DBD-MySQL -y

yum install perl-Time-HiRes -y

yum install perl-IO-Socket-SSL –y

yum install perl-TermReadKey.x86\_64 –y

MySQL使用xtrabackup备份时报错'Too many open files'

<http://blog.51cto.com/huangfuff/1610115>

./innobackupex --defaults-file=/etc/my.cnf --user=backup --password=mafei --socket=/opt/mysql/data/mysql.sock /opt/db\_backup/mysql\_bak/xtrabackup

**centos下编译安装xtrabackup2.4.6**

<https://blog.csdn.net/chenghuikai/article/details/68946617>

<http://downloads.sourceforge.net/project/boost/boost/1.59.0/boost_1_59_0.tar.gz>

安装boost

|  |
| --- |
| yum install cmake gcc gcc-c++ libaio libaio-devel automake autoconf bison libtool ncurses-devel libgcrypt-devel libev-devel libcurl-devel vim-common python-devel  tar xzf boost\_1\_59\_0.tar.gz  cd boost\_1\_59\_0  ./bootstrap.sh  ./b2 install --perfix=/usr/local/boost\_1\_59\_0 |

cmake -DDOWNLOAD\_BOOST=1 -DWITH\_BOOST=/usr/local/boost\_1\_59\_0 -DBUILD\_CONFIG=xtrabackup\_release -DWITH\_MAN\_PAGES=OFF && make -j4