## Mysql 备份还原说明

1. **xtrabackup**
   1. 安装xtrabackup

xtrabackup仅支持linux操作系统。

系统版本：CentOS 5，CentOS 6，CentOS 7

RHEL 5, RHEL 6, RHEL 7

xtrabackup版本：

2.4.10

安装：

1. 从yum源安装

|  |
| --- |
| yum install http://www.percona.com/downloads/percona-release/redhat/0.1-4/percona-release-0.1-4.noarch.rpm |

1. rpm安装

|  |
| --- |
| $ wget https://www.percona.com/downloads/XtraBackup/Percona-XtraBackup-2.4.4/**\** binary/redhat/7/x86\_64/percona-xtrabackup-24-2.4.4-1.el7.x86\_64.rpm  $ yum localinstall percona-xtrabackup-24-2.4.4-1.el7.x86\_64.rpm |

1. **Mysql相关配置**

Mysql用户授权。建立备份专用用户，并授予相应的权限。

|  |
| --- |
| mysql> CREATE USER 'bkpuser'@'localhost' IDENTIFIED BY 's3cret';  mysql> GRANT RELOAD, LOCK TABLES, PROCESS, REPLICATION CLIENT ON \*.\* TO 'bkpuser'@'localhost';  mysql> FLUSH PRIVILEGES; |

Mysql配置文件。5.7版本的Mysql不再提供my-default.cnf文件。Mysql不需要my.cnf也可以正常运行。但是使用xtrabackup备份数据库需要修改Mysql的相关配置，因此，手动创建my.cnf（附录A）并放置于/etc/目录下，并修改其权限：

chmod 777 /etc/my.cnf

1. **使用xtrabackup备份/还原**

3.1 修改脚本

在备份工作进行前，需要设定目标数据库的连接信息，修改备份脚本中的USER及PASSWORD为mysql的用户名及用户密码。BASEDIR为备份文件的存放路径。

执行备份脚本（附录B），即可生成备份文件。备份脚本中自动判断增量备份或是全量备份。每次增量备份都是基于前一次增量的基础上进行的。备份的过程会产生日志并保存在backup\_log文件夹下。

3.2 准备备份目录

在BASEDIR目录下新建以下几个目录：increment、full、backup\_log，拷贝backup.sh、recovery.sh到BASEDIR下，并修改其读写权限。

3.3 设置备份策略

备份策略默认为：周一到周六执行增量备份，周日执行全量备份。执行时间设置方法：

1. 打开cron任务编辑器：crontab –e
2. 输入调度策略，如每天0点执行backup.sh：

0 0 \* \* \* backup.sh

保存并推出编辑器。

c 查看调度任务：crontab –l，确认任务添加成功。

3.4 调试

执行备份命令进行调试：sh backup.sh，sh recovery.sh。

注意：

还原脚本仅支持本机还原，其原理为直接拷贝备份文件至mysql/data下实现。官方称远程还原功能带有BUG。

附录A my.cnf（黄色字体为修改部分）

# Example MySQL config file for small systems.

#

# This is for a system with little memory (<= 64M) where MySQL is only used

# from time to time and it's important that the mysqld daemon

# doesn't use much resources.

#

# MySQL programs look for option files in a set of

# locations which depend on the deployment platform.

# You can copy this option file to one of those

# locations. For information about these locations, see:

# http://dev.mysql.com/doc/mysql/en/option-files.html

#

# In this file, you can use all long options that a program supports.

# If you want to know which options a program supports, run the program

# with the "--help" option.

# The following options will be passed to all MySQL clients

[client]

default-character-set=utf8

#password = your\_password

port = 3306

socket = /tmp/mysql.sock

# Here follows entries for some specific programs

[xtrabackup]

target\_dir = /Users/limeng/temp/mysqlbak

# The MySQL server

[mysqld]

datadir=/usr/local/mysql/data/

default-storage-engine=INNODB

character-set-server=utf8

collation-server=utf8\_general\_ci

port = 3306

socket = /tmp/mysql.sock

skip-external-locking

key\_buffer\_size = 16K

max\_allowed\_packet = 1M

table\_open\_cache = 4

sort\_buffer\_size = 64K

read\_buffer\_size = 256K

read\_rnd\_buffer\_size = 256K

net\_buffer\_length = 2K

thread\_stack = 512K

secure\_file\_priv = /Users/limeng/temp/mysqlbak

# Don't listen on a TCP/IP port at all. This can be a security enhancement,

# if all processes that need to connect to mysqld run on the same host.

# All interaction with mysqld must be made via Unix sockets or named pipes.

# Note that using this option without enabling named pipes on Windows

# (using the "enable-named-pipe" option) will render mysqld useless!

#

#skip-networking

server-id = 1

# Uncomment the following if you want to log updates

#log-bin=mysql-bin

# binary logging format - mixed recommended

#binlog\_format=mixed

# Causes updates to non-transactional engines using statement format to be

# written directly to binary log. Before using this option make sure that

# there are no dependencies between transactional and non-transactional

# tables such as in the statement INSERT INTO t\_myisam SELECT \* FROM

# t\_innodb; otherwise, slaves may diverge from the master.

#binlog\_direct\_non\_transactional\_updates=TRUE

# Uncomment the following if you are using InnoDB tables

#innodb\_data\_home\_dir = /usr/local/mysql/data

#innodb\_data\_file\_path = ibdata1:10M:autoextend

#innodb\_log\_group\_home\_dir = /usr/local/mysql/data

# You can set ..\_buffer\_pool\_size up to 50 - 80 %

# of RAM but beware of setting memory usage too high

#innodb\_buffer\_pool\_size = 16M

#innodb\_additional\_mem\_pool\_size = 2M

# Set ..\_log\_file\_size to 25 % of buffer pool size

innodb\_log\_file\_size = 5M

#innodb\_log\_buffer\_size = 8M

#innodb\_flush\_log\_at\_trx\_commit = 1

#innodb\_lock\_wait\_timeout = 50

[mysqldump]

quick

max\_allowed\_packet = 16M

[mysql]

no-auto-rehash

# Remove the next comment character if you are not familiar with SQL

#safe-updates

[myisamchk]

key\_buffer\_size = 8M

sort\_buffer\_size = 8M

[mysqlhotcopy]

interactive-timeout

附录B 备份脚本 （黄色字体为修改部分）

#!/bin/bash

USER=root

PASSWORD=limeng

HOST=10.6.240.222

PORT=3306

DATABASE=HN\_TEST

BASEDIR=/Users/limeng/temp/mysqlbak

WK=`date +%u`

LOGFILE=$BASEDIR/backup\_log/`date +%Y-%m-%d`.log

# 判断是否进行过全量备份

count=`ls -l full/ |grep "^d"|wc -l`

echo $count

if [[ $WK -eq 7 || $count -eq 0 ]];then

# 全量备份

echo "FULL BACKUP MODE"

date > $LOGFILE

echo "Begin full-backup-------------------------------" >> $LOGFILE

innobackupex --user=$USER --password=$PASSWORD --host=$HOST --port=$PORT --databases=$DATABASE $BASEDIR/full >> $LOGFILE 2>&1

echo "end backup-------------------------------" >> $LOGFILE

date >> $LOGFILE

echo "Backup finished!"

else

# 增量备份

echo "INCREMENT BACKUP MODE"

cd $BASEDIR/full

FULLFILE=`ls -l | tail -n 1 | awk '{print $9}'`

date > $LOGFILE

echo "Begin incre-backup-------------------------------" >> $LOGFILE

innobackupex --user=$USER --password=$PASSWORD --host=$HOST --port=$PORT --database=$DATABASE --incremental-basedir=$BASEDIR/full/$FULLFILE --incremental $BASEDIR/increment

echo "end backup-------------------------------" >> $LOGFILE

date >> $LOGFILE

echo "Backup finished!"

fi

exit 0

附录C 还原脚本

#!/bin/bash

USER=root

PASSWORD=limeng

HOST=10.6.240.222

PORT=3306

DATABASE=HN\_TEST

BASEDIR=/Users/limeng/temp/mysqlbak

LOGFILE=$BASEDIR/recovery\_log/`date +%Y-%m-%d`.log

MYSQL\_DATA\_DIR=/usr/local/mysql/data

/etc/init.d/mysql stop

cd $BASEDIR/increment

DIR=`ls -l | tail -n 1 | awk '{print $9}'`

BAKDIR=$BASEDIR/increment/$BAKDIR

echo "start recover-------------------------------" >> $LOGFILE

innobackupex --apply-log --defaults-file=/etc/my.cnf --user=$USER --password=$PASSWORD $BAKDIR

rm -rf $MYSQL\_DATA\_DIR/$DATABASE

rm -f $MYSQL\_DATA\_DIR/ibdata1

rm -f $MYSQL\_DATA\_DIR/ib\_logfile0

rm -f $MYSQL\_DATA\_DIR/ib\_logfile1

cd /data/mysql/backups

tar zcvf `date +%Y-%m-%d`.tgz `date +%Y-%m-%d`

rm -rf `date -d -30day +%Y-%m-%d`.tgz

cp -r $BAKDIR/ib\* $MYSQL\_DATA\_DIR/

cp -r $BAKDIR/$DATABASE $MYSQL\_DATA\_DIR/

rm -rf $BAKDIR/\*

chown -R mysql.mysql $MYSQL\_DATA\_DIR/$DATABASE

chown mysql.mysql $MYSQL\_DATA\_DIR/ib\*

echo "end recover-------------------------------" >> $LOGFILE

/etc/init.d/mysql start