<https://www.cnblogs.com/waynechou/p/xtrabackup_backup.html>

<http://www.toxingwang.com/database/mysql/1539.html>

**1、介绍**

**1) innobackupex 参数选项**

--defaults-file=[MY.CNF]    //指定配置文件：只能从给定的文件中读取默认选项。 且必须作为命令行上的第一个选项；必须是一个真实的文件，它不能是一个符号链接。

--databases=#    //指定备份的数据库和表，格式为：--database="db1[.tb1] db2[.tb2]" 多个库之间以空格隔开，如果此选项不被指定，将会备份所有的数据库。

--include=REGEXP    //用正则表达式的方式指定要备份的数据库和表，格式为 --include=‘^mydb[.]mytb’ ，对每个库中的每个表逐一匹配，因此会创建所有的库，不过是空的目录。--include 传递给 xtrabackup --tables。

--tables-file=FILE    //此选项的参数需要是一个文件名，此文件中每行包含一个要备份的表的完整名称，格式为databasename.tablename。该选项传递给 xtrabackup --tables-file，与--tables选项不同，只有要备份的表的库才会被创建。

注意：部分备份（--include、--tables-file、--database）需要开启 innodb\_file\_per\_table 。

--compact    //创建紧凑型备份，忽略所有辅助索引页，只备份data page；通过--apply-log中重建索引--rebuild-indexs。

--compress    //此选项指示xtrabackup压缩备份的InnoDB数据文件，会生成 \*.qp 文件。

--decompress    //解压缩qp文件，为了解压缩，必须安装 qpress 工具。 Percona XtraBackup不会自动删除压缩文件，为了清理备份目录，用户应手动删除 \* .qp文件：find /data/backup -name "\*.qp" | xargs rm。

--no-timestamp    //指定了这个选项备份将会直接存储在 BACKUP-DIR 目录，不再创建时间戳文件夹。

--apply-log    //应用 BACKUP-DIR 中的 xtrabackup\_logfile 事务日志文件。一般情况下，在备份完成后，数据尚且不能用于恢复操作，因为备份的数据中可能会包含尚未提交的事务或已经提交但尚未同步至数据文件中的事务。因此，此时数据文件仍处于不一致状态。“准备”的主要作用正是通过回滚未提交的事务及同步已经提交的事务至数据文件使得数据文件处于一致性状态。

--use-memory=#    //此选项接受一个字符参数（1M/1MB,1G/1GB，默认100M），仅与--apply-log一起使用，该选项指定prepare时用于崩溃恢复（crash-recovery）的内存。

--copy-back    //拷贝先前备份所有文件到它们的原始路径。但原路径下不能有任何文件或目录，除非指定 --force-non-empty-directories 选项。

--force-non-empty-directories    //恢复时指定此选项，可使 --copy-back 和 --move-back 复制文件到非空目录，即原data目录下可以有其他文件，但是不能有与恢复文件中同名的文件，否则恢复失败。

--rsync    //此选项可优化本地文件（非InnoDB）的传输。rsync工具一次性拷贝所有非InnoDB文件，而不是为每个文件单独创建cp，在备份恢复很多数据库和表时非常高效。此选项不能和 --stream 一起使用。

--incremental    //这个选项告诉 xtrabackup 创建一个增量备份，而不是完全备份。它传递到 xtrabackup 子进程。当指定这个选项，可以设置 --incremental-lsn 或 --incremental-basedir。如果这2个选项都没有被指定，--incremental-basedir 传递给 xtrabackup 默认值，默认值为：基础备份目录的第一个时间戳备份目录。

--incremental-basedir=DIRECTORY    //该选项接受一个字符串参数，该参数指定作为增量备份的基本数据集的完整备份目录。它与 --incremental 一起使用。

--incremental-dir=DIRECTORY    //该选项接受一个字符串参数，该参数指定了增量备份将与完整备份相结合的目录，以便进行新的完整备份。它与 --incremental 选项一起使用。

--redo-only    //在“准备基本完整备份” 和 “合并所有的增量备份(除了最后一个增备)”时使用此选项。它直接传递给xtrabackup的 xtrabackup --apply-log-only 选项，使xtrabackup跳过"undo"阶段，只做"redo"操作。如果后面还有增量备份应用到这个全备,这是必要的。有关详细信息,请参阅xtrabackup文档。

--parallel=NUMBER-OF-THREADS    //此选项接受一个整数参数，指定xtrabackup子进程应用于同时备份文件的线程数。请注意，此选项仅适用于文件级别，也就是说，如果您有多个.ibd文件，则它们将被并行复制； 如果您的表一起存储在一个表空间文件中，它将不起作用。

**2) xtrabackup 参数选项**

--apply-log-only    //这个选项使在准备备份(prepare)时，只执行重做(redo)阶段，这对于增量备份非常重要。

2、安装

rpm -ivh 可能出现差包 用Yum安装

3、测试

3、1 全备

innobackupex --defaults-file=/etc/my.cnf --user=root --password=123456 /data/backup/

innobackupex: Backup created in directory '/data/backup/2018-04-06\_20-56-39'

innobackupex: MySQL binlog position: filename 'mysql-bin.000001', position 120

180406 20:56:48 innobackupex: Connection to database server closed

180406 20:56:48 innobackupex: completed OK!

全备目录中的文件

[root@redhat7 2018-04-06\_20-56-39]# ll

total 77920

-rw-r--r-- 1 root root 295 Apr 6 20:56 backup-my.cnf 备份命令用到的配置选项信息

ib\_buffer\_pool， buffer pool 中的热数据，当设置 innodb\_buffer\_pool\_dump\_at\_shutdown=1 ，在关闭 MySQL 时，会把内存中的热数据保存在磁盘里 ib\_buffer\_pool 文件中，位于数据目录下。

-rw-r----- 1 root root 79691776 Apr 6 20:56 ibdata1

drwx------ 2 root root 65536 Apr 6 20:56 mysql

drwxr-xr-x 2 root root 4096 Apr 6 20:56 performance\_schema

drwx------ 2 root root 4096 Apr 6 20:56 test

-rw-r--r-- 1 root root 21 Apr 6 20:56 xtrabackup\_binlog\_info 当前日志文件和位置

-rw-r----- 1 root root 89 Apr 6 20:56 xtrabackup\_checkpoints

备份类型（如完全或增量）、备份状态（如是否已经为prepared状态）和LSN(日志序列号)范围信息；

-rw-r--r-- 1 root root 584 Apr 6 20:56 xtrabackup\_info

记录备份的基本信息，uuid、备份命令、备份时间、binlog、LSN、以及其他加密压缩等信息

-rw-r----- 1 root root 2560 Apr 6 20:56 xtrabackup\_logfile

备份的重做日志文件。

3.2 全备恢复测试

由于第一次全备的数据可能含有一些未提交的事务，所以使用apply-log回滚未提交的事务及同步已经提交的事务至数据文件使得数据文件处于一致性状态。

innobackupex --apply-log /data/backup/2018-04-06\_20-56-39/

mv data data2109

service mysqld stop

恢复

innobackupex --defaults-file=/etc/my.cnf --copy-back --rsync /data/backup/2018-04-06\_20-56-39/

innobackupex: Starting to copy InnoDB log files

innobackupex: in '/data/backup/2018-04-06\_20-56-39'

innobackupex: back to original InnoDB log directory '/data/data'

innobackupex: Copying '/data/backup/2018-04-06\_20-56-39/ib\_logfile0' to '/data/data/ib\_logfile0'

innobackupex: Copying '/data/backup/2018-04-06\_20-56-39/ib\_logfile1' to '/data/data/ib\_logfile1'

innobackupex: Finished copying back files.

180406 21:11:21 innobackupex: completed OK!

更改数据目录权限和属主并启动

chown -R mysql:mysql /data/data

service mysqld start

4、增量备份、全量合并、增量恢复测试。

4.1 增量备份

建表并添加数据

mysql> create table test(

-> `id` int,

-> `name` varchar(20));

Query OK, 0 rows affected (0.04 sec)

mysql> show create table test;

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

| Table | Create Table |

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

| test | CREATE TABLE `test` (

`id` int(11) DEFAULT NULL,

`name` varchar(20) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8 |

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

1 row in set (0.00 sec)

mysql> insert into test values(1,'123456'),(2,'234566');

Query OK, 2 rows affected (0.00 sec)

Records: 2 Duplicates: 0 Warnings: 0

mysql> select \* from test;

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

| id | name |

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

| 1 | 123456 |

| 2 | 234566 |

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

2 rows in set (0.00 se

4.2 增量备份

innobackupex --defaults-file=/etc/my.cnf --user=root --password=123456 --incremental /data/backup/ --incremental-basedir=/data/backup/2018-04-06\_21-28-38/可以加--parallel=2

innobackupex: Backup created in directory '/data/backup/2018-04-06\_21-21-27'

innobackupex: MySQL binlog position: filename 'mysql-bin.000004', position 1603

180406 21:21:35 innobackupex: Connection to database server closed

180406 21:21:35 innobackupex: completed OK!

4.3 做一些改变

添加数据

mysql> insert into test select \*from test;

Query OK, 2 rows affected (0.08 sec)

Records: 2 Duplicates: 0 Warnings: 0

mysql> select \*from test;

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

| id | name |

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

| 1 | 123456 |

| 2 | 234566 |

| 1 | 123456 |

| 2 | 234566 |

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

4 rows in set (0.00 sec)

4.4 二次增量备份

以一次增量为基础

innobackupex --defaults-file=/etc/my.cnf --user=root --password=123456 --incremental /data/backup/ --incremental-basedir=/data/backup/2018-04-06\_21-30-43/

innobackupex: Backup created in directory '/data/backup/2018-04-06\_21-31-42'

innobackupex: MySQL binlog position: filename 'mysql-bin.000004', position 1831

180406 21:31:49 innobackupex: Connection to database server closed

180406 21:31:49 innobackupex: completed OK!

4.5 在做一些修改

mysql> select \*from test;

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

| id | name |

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

| 1 | 123456 |

| 2 | 234566 |

| 1 | 123456 |

| 2 | 234566 |

| 1 | 123456 |

| 2 | 234566 |

| 1 | 123456 |

| 2 | 234566 |

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

8 rows in set (0.00 sec)

4.6 恢复

1 准备全备

innobackupex --apply-log --redo-only /data/backup/2018-04-06\_21-28-38/

[notice (again)]

If you use binary log and don't use any hack of group commit,

the binary log position seems to be:

InnoDB: Last MySQL binlog file position 0 1831, file name mysql-bin.000004

xtrabackup: starting shutdown with innodb\_fast\_shutdown = 1

InnoDB: Starting shutdown...

InnoDB: Shutdown completed; log sequence number 2004933

180406 21:36:07 innobackupex: completed OK!

2、合并第一次增量

innobackupex --apply-log --redo-only /data/backup/2018-04-06\_21-28-38/ --incremental-dir=/data/backup/2018-04-06\_21-30-43/

rformance\_schema/table\_lock\_waits\_summary\_by\_table.frm'

180406 21:37:36 innobackupex: completed OK!

3、合并第二次增量 最后一次增量不要添加 --redo-only参数

innobackupex --apply-log /data/backup/2018-04-06\_21-28-38/ --incremental-dir=/data/backup/2018-04-06\_21-31-42/

rformance\_schema/table\_lock\_waits\_summary\_by\_table.frm'

180406 21:38:51 innobackupex: completed OK

4、准备恢复

service mysqld stop

mv data data2139

innobackupex --defaults-file=/etc/my.cnf --copy-back --rsync /data/backup/2018-04-06\_21-28-38/

innobackupex: Starting to copy InnoDB log files

innobackupex: in '/data/backup/2018-04-06\_21-28-38'

innobackupex: back to original InnoDB log directory '/data/data'

innobackupex: Finished copying back files.

180406 21:41:07 innobackupex: completed OK!

更改属主

chown -R mysql:mysql /data/data

启动服务

service mysqld start

5、后续的修改添加进去

[root@redhat7 2018-04-06\_21-31-42]# cat xtrabackup\_binlog\_info

mysql-bin.000004 1831

mysqlbinlog --start-position=1831 /data/data/mysql-bin.000004 >/data/test.sql

[root@redhat7 data]# mysql -u root -p test </data/test.sql

Enter password:

[root@redhat7 data]# mysql -u root -p

Enter password:

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 15

Server version: 5.6.19-log Source distribution

Copyright (c) 2000, 2014, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its

affiliates. Other names may be trademarks of their respective

owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use test;

Database changed

mysql> select \* from test;

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

| id | name |

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

| 1 | 123456 |

| 2 | 234566 |

| 1 | 123456 |

| 2 | 234566 |

| 1 | 123456 |

| 2 | 234566 |

| 1 | 123456 |

| 2 | 234566 |

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

8 rows in set (0.00 sec)

6 做一个全新的全备

innobackupex --defaults-file=/etc/my.cnf --user=root --password=123456 /data/backup/

innobackupex: Backup created in directory '/data/backup/2018-04-06\_21-45-36'

innobackupex: MySQL binlog position: filename 'mysql-bin.000001', position 349

180406 21:45:44 innobackupex: Connection to database server closed

180406 21:45:44 innobackupex: completed OK!

其实之前的备份可以删除掉了