

熟悉mydumper myloader

mydumper特性

- 多线程备份，生成多个备份文件
- 针对MyISAM表加读锁，会阻塞DML语句
- 保证备份数据一致性
- 支持文件压缩和导出binlog
- 支持多线程恢复
- 支持守护进程模式，定时快照和连续二进制日志
- 支持正则匹配
- 恢复支持是否启用binlog及指定事物的大小(多少insert)
- 无法备份视图和触发器
- 备份整库时可用备份存储过程和函数，单库备份等情况无法对其进行备份

mydumper备份机制

主要步骤概括

1. 主线程 FLUSH TABLES WITH READ LOCK, 施加全局只读锁，以阻止DML语句写入，保证数据的一致性
2. 读取当前时间点的二进制日志文件名和日志写入的位置并记录在metadata文件中，以供即使点恢复使用
3. N个（线程数可以指定，默认是4）dump线程 START TRANSACTION WITH CONSISTENT SNAPSHOT; 开启读一致的事物
4. dump non-InnoDB tables, 首先导出非事物引擎的表
5. 主线程 UNLOCK TABLES 非事物引擎备份完后，释放全局只读锁
6. dump InnoDB tables, 基于事物导出InnoDB表
7. 事物结束

备份所生成的文件

1. 所有的备份文件在一个目录中，目录可以自己指定
2. 目录中包含一个metadata文件

记录了备份数据库在备份时间点的二进制日志文件名，日志的写入位置，
如果是在从库进行备份，还会记录备份时同步至从库的二进制日志文件及写入位置

3. 每个表有两个备份文件：

database.table-schema.sql 表结构文件

database.table.sql 表数据文件

如果对表文件分片，将生成多个备份数据文件，可以指定行数或指定大小分片

mydumper help信息：

```
[root@MysqlRestore ~]# mydumper --help
```

Usage:

mydumper [OPTION...] multi-threaded MySQL dumping

Help Options:

-?, --help Show help options

Application Options:

-B, --database	Database to dump
-T, --tables-list	Comma delimited table list to dump (does not exclude regex option)
-o, --outputdir	Directory to output files to
-s, --statement-size	Attempted size of INSERT statement in bytes, default 1000000
-r, --rows	Try to split tables into chunks of this many rows. This option turns off --chunk-filesize
-F, --chunk-filesize	Split tables into chunks of this output file size. This value is in MB
-c, --compress	Compress output files
-e, --build-empty-files	Build dump files even if no data available from table
-x, --regex	Regular expression for 'db.table' matching
-i, --ignore-engines	Comma delimited list of storage engines to ignore
-m, --no-schemas	Do not dump table schemas with the data
-k, --no-locks	Do not execute the temporary shared read lock. WARNING: This will cause inconsistent backups

```

--less-locking           Minimize locking time on InnoDB tables.
-l, --long-query-guard   Set long query timer in seconds, default 60
-K, --kill-long-queries  Kill long running queries (instead of aborting)
-D, --daemon             Enable daemon mode
-I, --snapshot-interval  Interval between each dump snapshot (in minutes), requires --daemon, default 60
-L, --logfile            Log file name to use, by default stdout is used
--tz-utc                 SET TIME_ZONE='+00:00' at top of dump to allow dumping of TIMESTAMP data when a server has data in
different time zones or data is being moved between servers with different time zones, defaults to on use --skip-tz-utc to
disable.
--skip-tz-utc
--use-savepoints          Use savepoints to reduce metadata locking issues, needs SUPER privilege
--success-on-1146        Not increment error count and Warning instead of Critical in case of table doesn't exist
--lock-all-tables       Use LOCK TABLE for all, instead of FTWRL
-h, --host               The host to connect to
-u, --user               Username with privileges to run the dump
-p, --password           User password
-P, --port               TCP/IP port to connect to
-S, --socket             UNIX domain socket file to use for connection
-t, --threads            Number of threads to use, default 4
-C, --compress-protocol  Use compression on the MySQL connection
-V, --version            Show the program version and exit
-v, --verbose            Verbosity of output, 0 = silent, 1 = errors, 2 = warnings, 3 = info, default 2

```

[root@MysqlRestore ~]#

myloader help信息 :

[root@MysqlRestore ~]# myloader --help

Usage:

myloader [OPTION...] multi-threaded MySQL loader

Help Options:

-?, --help Show help options

Application Options:

```

-d, --directory           Directory of the dump to import
-q, --queries-per-transaction  Number of queries per transaction, default 1000
-o, --overwrite-tables    Drop tables if they already exist
-B, --database            An alternative database to restore into
-e, --enable-binlog       Enable binary logging of the restore data
-h, --host               The host to connect to
-u, --user               Username with privileges to run the dump
-p, --password           User password
-P, --port               TCP/IP port to connect to
-S, --socket             UNIX domain socket file to use for connection
-t, --threads            Number of threads to use, default 4
-C, --compress-protocol  Use compression on the MySQL connection
-V, --version            Show the program version and exit
-v, --verbose            Verbosity of output, 0 = silent, 1 = errors, 2 = warnings, 3 = info, default 2

```

[root@MysqlRestore ~]#

参考 : <http://www.simlinux.com/archives/470.html>

<http://blog.csdn.net/miyatang/article/details/60875086>

mydump 安装步骤 :

yum search wget

yum install -y wget

wget <https://launchpad.net/mydumper/0.6/0.6.2/+download/mydumper-0.6.2.tar.gz>

```
yum search cmake
yum install -y cmake
```

eg :

备份全库 :

```
mydumper -u root -p passwd -S /path/to/socket -o /path/to/output/file
```

安装报错 :

```
[root@MysqlRestore mydumper-0.6.2]#
[root@MysqlRestore mydumper-0.6.2]# cmake .
-- The C compiler identification is unknown
-- The CXX compiler identification is unknown
CMake Error: your C compiler: "CMAKE_C_COMPILER-NOTFOUND" was not found. Please set CMAKE_C_COMPILER to a valid compiler path or name.
CMake Error: your CXX compiler: "CMAKE_CXX_COMPILER-NOTFOUND" was not found. Please set CMAKE_CXX_COMPILER to a valid compiler path or name.
-- MySQL not found.
-- Could NOT find ZLIB (missing: ZLIB_LIBRARY ZLIB_INCLUDE_DIR)
-- Found PkgConfig: /usr/bin/pkg-config (found version "0.27.1")
-- checking for one of the modules 'glib-2.0'
CMake Error at /usr/share/cmake/Modules/FindPkgConfig.cmake:363 (message):
  None of the required 'glib-2.0' found
Call Stack (most recent call first):
  cmake/modules/FindGLIB2.cmake:10 (pkg_search_module)
  CMakeLists.txt:10 (find_package)

-- checking for one of the modules 'gthread-2.0'
CMake Error at /usr/share/cmake/Modules/FindPkgConfig.cmake:363 (message):
  None of the required 'gthread-2.0' found
Call Stack (most recent call first):
  cmake/modules/FindGLIB2.cmake:11 (pkg_search_module)
  CMakeLists.txt:10 (find_package)

-- checking for module 'libpcre'
-- found libpcre, version 8.32
-- Found PCRE: /usr/include

CMake Warning at docs/CMakeLists.txt:9 (message):
  Unable to find Sphinx documentation generator

-----
-- MYSQL_CONFIG = MYSQL_CONFIG-NOTFOUND
-- CMAKE_INSTALL_PREFIX = /usr/local
-- BUILD_DOCS = ON
-- WITH_BINLOG = OFF
-- RUN_CPPCHECK = OFF
-- Change a values with: cmake -D<Variable>=<Value>
```

CMake Error: The following variables are used in this project, but they are set to NOTFOUND.

Please set them or make sure they are set and tested correctly in the CMake files:

GLIB2_LIBRARIES (ADVANCED)

linked by target "mydumper" in directory /srv/download/mydumper-0.6.2

linked by target "myloader" in directory /srv/download/mydumper-0.6.2

GTHREAD2_LIBRARIES (ADVANCED)

linked by target "mydumper" in directory /srv/download/mydumper-0.6.2

linked by target "myloader" in directory /srv/download/mydumper-0.6.2

MYSQL_INCLUDE_DIR (ADVANCED)

used as include directory in directory /srv/download/mydumper-0.6.2

used as include directory in directory /srv/download/mydumper-0.6.2

used as include directory in directory /srv/download/mydumper-0.6.2

MYSQL_LIBRARIES_mysqlclient

linked by target "mydumper" in directory /srv/download/mydumper-0.6.2

linked by target "myloader" in directory /srv/download/mydumper-0.6.2

-- Configuring incomplete, errors occurred!

See also "/srv/download/mydumper-0.6.2/CMakeFiles/CMakeOutput.log".

See also "/srv/download/mydumper-0.6.2/CMakeFiles/CMakeError.log".

[root@MysqlRestore mydumper-0.6.2]#

解决方法：

yum -y install gcc

yum -y install gcc-c++ kbase-workspace-devel dbusmenu-qt-devel

[root@MysqlRestore mydumper-0.6.2]# cmake .

-- The CXX compiler identification is GNU 4.8.5

-- Check for working CXX compiler: /usr/bin/c++

-- Check for working CXX compiler: /usr/bin/c++ -- works

-- Detecting CXX compiler ABI info

-- Detecting CXX compiler ABI info - done

-- MySQL not found.

-- Found ZLIB: /usr/lib64/libz.so (found version "1.2.7")

-- checking for one of the modules 'glib-2.0'

CMake Error at /usr/share/cmake/Modules/FindPkgConfig.cmake:363 (message):

None of the required 'glib-2.0' found

Call Stack (most recent call first):

cmake/modules/FindGLIB2.cmake:10 (pkg_search_module)

CMakelists.txt:10 (find_package)

-- checking for one of the modules 'gthread-2.0'

CMake Error at /usr/share/cmake/Modules/FindPkgConfig.cmake:363 (message):

None of the required 'gthread-2.0' found

Call Stack (most recent call first):

cmake/modules/FindGLIB2.cmake:11 (pkg_search_module)

CMakelists.txt:10 (find_package)

CMake Warning at docs/CMakelists.txt:9 (message):

Unable to find Sphinx documentation generator

```
-- MYSQL_CONFIG = MYSQL_CONFIG-NOTFOUND
-- CMAKE_INSTALL_PREFIX = /usr/local
-- BUILD_DOCS = ON
-- WITH_BINLOG = OFF
-- RUN_CPPCHECK = OFF
-- Change a values with: cmake -D<Variable>=<Value>
```

CMake Error: The following variables are used in this project, but they are set to NOTFOUND.

Please set them or make sure they are set and tested correctly in the CMake files:

GLIB2_LIBRARIES (ADVANCED)

linked by target "mydumper" in directory /srv/download/mydumper-0.6.2

linked by target "myloader" in directory /srv/download/mydumper-0.6.2

GTHREAD2_LIBRARIES (ADVANCED)

linked by target "mydumper" in directory /srv/download/mydumper-0.6.2

linked by target "myloader" in directory /srv/download/mydumper-0.6.2

MYSQL_INCLUDE_DIR (ADVANCED)

used as include directory in directory /srv/download/mydumper-0.6.2

used as include directory in directory /srv/download/mydumper-0.6.2

used as include directory in directory /srv/download/mydumper-0.6.2

MYSQL_LIBRARIES_mysqlclient

linked by target "mydumper" in directory /srv/download/mydumper-0.6.2

linked by target "myloader" in directory /srv/download/mydumper-0.6.2

-- Configuring incomplete, errors occurred!

See also "/srv/download/mydumper-0.6.2/CMakeFiles/CMakeOutput.log".

See also "/srv/download/mydumper-0.6.2/CMakeFiles/CMakeError.log".

[root@MysqlRestore mydumper-0.6.2]#

解决方法：

Try to remove the directory of decompressed mydumper, then decompress the "mydumper-0.6.2.tar.gz" again, and cmake it, you'll find it works well.

Here is my operation:

```
rm -rf mydumper-0.6.2
```

```
tar -zxvf mydumper-0.6.2.tar.gz
```

```
cd mydumper-0.6.2
```

```
cmake .
```

```
make
```

```
make install
```

Good Luck!

参考：<https://stackoverflow.com/questions/28712175/cmake-is-not-possible-to-find-mysql-libraries-when-i-setup-mydumper>

```
[root@MysqlRestore mydumper-0.6.2]# locate mydumper
```

```
-bash: locate: command not found
```

```
[root@MysqlRestore mydumper-0.6.2]# whereis mydumper
```

```
mydumper: /usr/local/bin/mydumper
```

```
[root@MysqlRestore mydumper-0.6.2]#
```

```
[root@MysqlRestore mydumper-0.6.2]# whereis myloader
```

```
myloader: /usr/local/bin/myloader
```

```
[root@MysqlRestore mydumper-0.6.2]#
```

```
[root@MysqlRestore mydumper-0.6.2]# mydumper --version
```

```
mydumper 0.6.2, built against MySQL 5.7.19
```

```
[root@MysqlRestore mydumper-0.6.2]#
```

```
[root@MysqlRestore mydumper-0.6.2]# myloader --version
```

myloader 0.6.2, built against MySQL 5.7.19

[root@MysqlRestore mydumper-0.6.2]#

[root@MysqlRestore mydumper-0.6.2]#

Ok 安装完成

Over