NAME
mysqld_multi − manage multiple MySQL servers

SYNOPSIS
mysqld_multi [*options***] {start|stop|report}** [*GNR***[,***GNR***] ...]**

DESCRIPTION
mysqld_multi is designed to manage several **mysqld** processes that listen for connections on different Unix socket files and TCP/IP ports. It can start or stop servers, or report their current status.

> **Note**
> For some Linux platforms, MySQL installation from RPM or Debian packages includes systemd support for managing MySQL server startup and shutdown. On these platforms, **mysqld_multi** is not installed because it is unnecessary. For information about using systemd to handle multiple MySQL instances, see Section 2.5.9, "Managing MySQL Server with systemd".

mysqld_multi searches for groups named [mysqld*N*] in my.cnf (or in the file named by the **−−defaults−file** option). *N* can be any positive integer. This number is referred to in the following discussion as the option group number, or *GNR*. Group numbers distinguish option groups from one another and are used as arguments to **mysqld_multi** to specify which servers you want to start, stop, or obtain a status report for. Options listed in these groups are the same that you would use in the [mysqld] group used for starting **mysqld**. (See, for example, Section 2.10.5, "Starting and Stopping MySQL Automatically".) However, when using multiple servers, it is necessary that each one use its own value for options such as the Unix socket file and TCP/IP port number. For more information on which options must be unique per server in a multiple−server environment, see Section 5.8, "Running Multiple MySQL Instances on One Machine".

To invoke **mysqld_multi**, use the following syntax:

shell> **mysqld_multi [***options***] {start|stop|reload|report}** [*GNR***[,***GNR***] ...]**

start, stop, reload (stop and restart), and report indicate which operation to perform. You can perform the designated operation for a single server or multiple servers, depending on the *GNR* list that follows the option name. If there is no list, **mysqld_multi** performs the operation for all servers in the option file.

Each *GNR* value represents an option group number or range of group numbers. The value should be the number at the end of the group name in the option file. For example, the *GNR* for a group named [mysqld17] is 17. To specify a range of numbers, separate the first and last numbers by a dash. The *GNR* value 10−13 represents groups [mysqld10] through [mysqld13]. Multiple groups or group ranges can be specified on the command line, separated by commas. There must be no whitespace characters (spaces or tabs) in the *GNR* list; anything after a whitespace character is ignored.

This command starts a single server using option group [mysqld17]:

shell> **mysqld_multi start 17**

This command stops several servers, using option groups [mysqld8] and [mysqld10] through [mysqld13]:

shell> **mysqld_multi stop 8,10−13**

For an example of how you might set up an option file, use this command:

shell> **mysqld_multi −−example**

mysqld_multi searches for option files as follows:

- With **−−no−defaults**, no option files are read.

- With **−−defaults−file=***file_name*, only the named file is read.

- Otherwise, option files in the standard list of locations are read, including any file named by the **−−defaults−extra−file=***file_name* option, if one is given. (If the option is given multiple times, the last value is used.)

For additional information about these and other option−file options, see Section 4.2.2.3, "Command-Line Options that Affect Option-File Handling".

Option files read are searched for [mysqld_multi] and [mysqld*N*] option groups. The [mysqld_multi] group can be used for options to **mysqld_multi** itself. [mysqld*N*] groups can be used for options passed to specific **mysqld** instances.

The [mysqld] or [mysqld_safe] groups can be used for common options read by all instances of **mysqld** or **mysqld_safe**. You can specify a **−−defaults−file=***file_name* option to use a different configuration file for that instance, in which case the [mysqld] or [mysqld_safe] groups from that file will be used for that instance.

mysqld_multi supports the following options.

- **−−help**

  Display a help message and exit.

- **−−example**

  Display a sample option file.

- **−−log=***file_name*

  Specify the name of the log file. If the file exists, log output is appended to it.

- **−−mysqladmin=***prog_name*

  The **mysqladmin** binary to be used to stop servers.

- **−−mysqld=***prog_name*

  The **mysqld** binary to be used. Note that you can specify **mysqld_safe** as the value for this option also. If you use **mysqld_safe** to start the server, you can include the mysqld or ledir options in the corresponding [mysqld*N*] option group. These options indicate the name of the server that **mysqld_safe** should start and the path name of the directory where the server is located. (See the descriptions for these options in **mysqld_safe**(1).) Example:

  [mysqld38]
  mysqld = mysqld−debug
  ledir = /opt/local/mysql/libexec

- **−−no−log**

  Print log information to stdout rather than to the log file. By default, output goes to the log file.

- **−−password=***password*

  The password of the MySQL account to use when invoking **mysqladmin**. Note that the password value is not optional for this option, unlike for other MySQL programs.

- **−−silent**

  Silent mode; disable warnings.

- **−−tcp−ip**

  Connect to each MySQL server through the TCP/IP port instead of the Unix socket file. (If a socket file is missing, the server might still be running, but accessible only through the TCP/IP port.) By default, connections are made using the Unix socket file. This option affects stop and report operations.

- **−−user=**_user_name_

  The user name of the MySQL account to use when invoking **mysqladmin**.

- **−−verbose**

  Be more verbose.

- **−−version**

  Display version information and exit.

Some notes about **mysqld_multi**:

- **Most important**: Before using **mysqld_multi** be sure that you understand the meanings of the options that are passed to the **mysqld** servers and _why_ you would want to have separate **mysqld** processes. Beware of the dangers of using multiple **mysqld** servers with the same data directory. Use separate data directories, unless you _know_ what you are doing. Starting multiple servers with the same data directory does _not_ give you extra performance in a threaded system. See Section 5.8, "Running Multiple MySQL Instances on One Machine".

  Important

  Make sure that the data directory for each server is fully accessible to the Unix account that the specific **mysqld** process is started as. _Do not_ use the Unix _root_ account for this, unless you _know_ what you are doing. See Section 6.1.5, "How to Run MySQL as a Normal User".

- Make sure that the MySQL account used for stopping the **mysqld** servers (with the **mysqladmin** program) has the same user name and password for each server. Also, make sure that the account has the SHUTDOWN privilege. If the servers that you want to manage have different user names or passwords for the administrative accounts, you might want to create an account on each server that has the same user name and password. For example, you might set up a common multi_admin account by executing the following commands for each server:

  shell> **mysql −u root −S /tmp/mysql.sock −p**
  Enter password:
  mysql> **CREATE USER 'multi_admin'@'localhost' IDENTIFIED BY 'multipass';**
  mysql> **GRANT SHUTDOWN ON *.* TO 'multi_admin'@'localhost';**

  See Section 6.2, "Access Control and Account Management". You have to do this for each **mysqld** server. Change the connection parameters appropriately when connecting to each one. Note that the host name part of the account name must permit you to connect as multi_admin from the host where you want to run **mysqld_multi**.

- The Unix socket file and the TCP/IP port number must be different for every **mysqld**. (Alternatively, if the host has multiple network addresses, you can set the bind_address system variable to cause different servers to listen to different interfaces.)

- The **−−pid−file** option is very important if you are using **mysqld_safe** to start **mysqld** (for example, **−−mysqld=mysqld_safe**) Every **mysqld** should have its own process ID file. The advantage of using **mysqld_safe** instead of **mysqld** is that **mysqld_safe** monitors its **mysqld** process and restarts it if the process terminates due to a signal sent using kill −9 or for other reasons, such as a segmentation fault.

- You might want to use the **−−user** option for **mysqld**, but to do this you need to run the **mysqld_multi** script as the Unix superuser (root). Having the option in the option file doesn't matter; you just get a warning if you are not the superuser and the **mysqld** processes are started under your own Unix account.

The following example shows how you might set up an option file for use with **mysqld_multi**. The order in which the **mysqld** programs are started or stopped depends on the order in which they appear in the option

file. Group numbers need not form an unbroken sequence. The first and fifth [mysqld*N*] groups were intentionally omitted from the example to illustrate that you can have "gaps" in the option file. This gives you more flexibility.

```
# This is an example of a my.cnf file for mysqld_multi.
# Usually this file is located in home dir ˜/.my.cnf or /etc/my.cnf
[mysqld_multi]
mysqld     = /usr/local/mysql/bin/mysqld_safe
mysqladmin = /usr/local/mysql/bin/mysqladmin
user       = multi_admin
password   = my_password
[mysqld2]
socket     = /tmp/mysql.sock2
port       = 3307
pid−file   = /usr/local/mysql/data2/hostname.pid2
datadir    = /usr/local/mysql/data2
language   = /usr/local/mysql/share/mysql/english
user       = unix_user1
[mysqld3]
mysqld     = /path/to/mysqld_safe
ledir      = /path/to/mysqld−binary/
mysqladmin = /path/to/mysqladmin
socket     = /tmp/mysql.sock3
port       = 3308
pid−file   = /usr/local/mysql/data3/hostname.pid3
datadir    = /usr/local/mysql/data3
language   = /usr/local/mysql/share/mysql/swedish
user       = unix_user2
[mysqld4]
socket     = /tmp/mysql.sock4
port       = 3309
pid−file   = /usr/local/mysql/data4/hostname.pid4
datadir    = /usr/local/mysql/data4
language   = /usr/local/mysql/share/mysql/estonia
user       = unix_user3
[mysqld6]
socket     = /tmp/mysql.sock6
port       = 3311
pid−file   = /usr/local/mysql/data6/hostname.pid6
datadir    = /usr/local/mysql/data6
language   = /usr/local/mysql/share/mysql/japanese
user       = unix_user4
```

See Section 4.2.2.2, "Using Option Files".

COPYRIGHT

http://www.gnu.org/licenses/.

SEE ALSO

For more information, please refer to the MySQL Reference Manual, which may already be installed locally and which is also available online at http://dev.mysql.com/doc/.

AUTHOR

Oracle Corporation (http://dev.mysql.com/).