NAME

mysqld_safe - MySQL server startup script

SYNOPSIS

mysqld_safe *options*

DESCRIPTION

mysqld_safe is the recommended way to start a **mysqld** server on Unix. **mysqld_safe** adds some safety features such as restarting the server when an error occurs and logging runtime information to an error log. A description of error logging is given later in this section.

mysqld_safe tries to start an executable named **mysqld**. To override the default behavior and specify explicitly the name of the server you want to run, specify a **--mysqld** or **--mysqld-version** option to **mysqld_safe**. You can also use **--ledir** to indicate the directory where **mysqld_safe** should look for the server.

Many of the options to **mysqld_safe** are the same as the options to **mysqld**. See Section 5.1.6, "Server Command Options".

Options unknown to **mysqld_safe** are passed to **mysqld** if they are specified on the command line, but ignored if they are specified in the [mysqld_safe] group of an option file. See Section 4.2.6, "Using Option Files".

mysqld_safe reads all options from the [mysqld], [server], and [mysqld_safe] sections in option files. For example, if you specify a [mysqld] section like this, **mysqld_safe** will find and use the **--log-error** option:

[mysqld]
log-error=error.log

For backward compatibility, **mysqld_safe** also reads [safe_mysqld] sections, but to be current you should rename such sections to [mysqld_safe].

mysqld_safe supports the following options. It also reads option files and supports the options for processing them described at Section 4.2.7, "Command-Line Options that Affect Option-File Handling".

- **--help**

  Display a help message and exit.
- **--basedir=***dir_name*

  The path to the MySQL installation directory.
- **--core-file-size=***size*

  The size of the core file that **mysqld** should be able to create. The option value is passed to **ulimit -c**.
- **--datadir=***dir_name*

  The path to the data directory.
- **--defaults-extra-file=***file_name*

  Read this option file in addition to the usual option files. If the file does not exist or is otherwise inaccessible, the server will exit with an error. *file_name* is interpreted relative to the current directory if given as a relative path name rather than a full path name. This must be the first option on the command line if it is used.

  For additional information about this option, see Section 4.2.7, "Command-Line Options that Affect Option-File Handling".
- **--defaults-file=***file_name*

  Use only the given option file. If the file does not exist or is otherwise inaccessible, the server will exit with an error. *file_name* is interpreted relative to the current directory if given as a relative path name rather than a full path name. This must be the first option on the command line if it is used.

For additional information about this option, see Section 4.2.7, "Command-Line Options that Affect Option-File Handling".

- **--ledir=***dir_name*

  If **mysqld_safe** cannot find the server, use this option to indicate the path name to the directory where the server is located.

  As of MySQL 5.5.54, this option is accepted only on the command line, not in option files.

- **--log-error=***file_name*

  Write the error log to the given file. See Section 5.4.2, "The Error Log".

- **--malloc-lib=[***lib_name***]** The name of the library to use for memory allocation instead of the system malloc() library. As of MySQL 5.5.52, the option value must be one of the directories /usr/lib, /usr/lib64, /usr/lib/i386-linux-gnu, or /usr/lib/x86_64-linux-gnu. Prior to MySQL 5.5.52, any library can be used by specifying its path name, but there is a shortcut form to enable use of the tcmalloc library that is shipped with binary MySQL distributions for Linux in MySQL 5.5. It is possible that the shortcut form will not work under certain configurations, in which case you should specify a path name instead.

  Note

  As of MySQL 5.5.50, MySQL distributions no longer include a tcmalloc library.

The **--malloc-lib** option works by modifying the LD_PRELOAD environment value to affect dynamic linking to enable the loader to find the memory-allocation library when **mysqld** runs:

- If the option is not given, or is given without a value (**--malloc-lib=**), LD_PRELOAD is not modified and no attempt is made to use tcmalloc.
- If the option is given as **--malloc-lib=tcmalloc**, **mysqld_safe** looks for a tcmalloc library in /usr/lib and then in the MySQL pkglibdir location (for example, /usr/local/mysql/lib or whatever is appropriate). If tmalloc is found, its path name is added to the beginning of the LD_PRELOAD value for **mysqld**. If tcmalloc is not found, **mysqld_safe** aborts with an error.
- If the option is given as **--malloc-lib=***/path/to/some/library*, that full path is added to the beginning of the LD_PRELOAD value. If the full path points to a nonexistent or unreadable file, **mysqld_safe** aborts with an error.
- For cases where **mysqld_safe** adds a path name to LD_PRELOAD, it adds the path to the beginning of any existing value the variable already has.

Linux users can use the libtcmalloc_minimal.so included in binary packages by adding these lines to the my.cnf file:

[mysqld_safe]
malloc-lib=tcmalloc

Those lines also suffice for users on any platform who have installed a tcmalloc package in /usr/lib. To use a specific tcmalloc library, specify its full path name. Example:

[mysqld_safe]
malloc-lib=/opt/lib/libtcmalloc_minimal.so

- **--mysqld=***prog_name*

  The name of the server program (in the ledir directory) that you want to start. This option is needed if you use the MySQL binary distribution but have the data directory outside of the binary distribution. If **mysqld_safe** cannot find the server, use the **--ledir** option to indicate the path name to the directory where the server is located.

  As of MySQL 5.5.52, this option is accepted only on the command line, not in option files.

- **--mysqld-version=***suffix*

  This option is similar to the **--mysqld** option, but you specify only the suffix for the server program name. The base name is assumed to be **mysqld**. For example, if you use **--mysqld-version=debug**, **mysqld_safe** starts the **mysqld-debug** program in the ledir directory. If the argument to **--mysqld-version** is empty, **mysqld_safe** uses **mysqld** in

the ledir directory.

As of MySQL 5.5.52, this option is accepted only on the command line, not in option files.

- **--nice=***priority*

  Use the nice program to set the servers scheduling priority to the given value.

- **--no-defaults**

  Do not read any option files. If program startup fails due to reading unknown options from an option file, **--no-defaults** can be used to prevent them from being read. This must be the first option on the command line if it is used.

  For additional information about this option, see Section 4.2.7, "Command-Line Options that Affect Option-File Handling".

- **--open-files-limit=***count*

  The number of files that **mysqld** should be able to open. The option value is passed to **ulimit -n**.

  > **Note**
  >
  > You must start **mysqld_safe** as root for this to function properly.

- **--pid-file=***file_name*

  The path name that **mysqld** should use for its process ID file.

- **--plugin-dir=***dir_name*

  The path name of the plugin directory. This option was added in MySQL 5.5.3.

- **--port=***port_num*

  The port number that the server should use when listening for TCP/IP connections. The port number must be 1024 or higher unless the server is started by the root system user.

- **--skip-kill-mysqld**

  Do not try to kill stray **mysqld** processes at startup. This option works only on Linux.

- **--socket=***path*

  The Unix socket file that the server should use when listening for local connections.

- **--syslog**, **--skip-syslog**

  --syslog causes error messages to be sent to syslog on systems that support the **logger** program. --skip-syslog suppresses the use of syslog; messages are written to an error log file.

  When syslog is used, the daemon.err syslog facility/severity is used for all log messages.

  mysqld_safe ignores **--syslog** if **--log-error** is also given.

- **--syslog-tag=***tag*

  For logging to syslog, messages from **mysqld_safe** and **mysqld** are written with identifiers of mysqld_safe and mysqld, respectively. To specify a suffix for the identifiers, use **--syslog-tag=***tag*, which modifies the identifiers to be mysqld_safe-*tag* and mysqld-*tag*.

- **--timezone=***timezone*

  Set the TZ time zone environment variable to the given option value. Consult your operating system documentation for legal time zone specification formats.

- **--user=**{*user_name*|*user_id*}

  Run the **mysqld** server as the user having the name *user_name* or the numeric user ID *user_id*. ("User" in this context refers to a system login account, not a MySQL user listed in the grant tables.)

If you execute **mysqld_safe** with the **--defaults-file** or **--defaults-extra-file** option to name an option file, the option must be the first one given on the command line or the option file will not be used. For example, this command will not use the named option file:

mysql> **mysqld_safe --port=***port_num* **--defaults-file=***file_name*

Instead, use the following command:

mysql> **mysqld_safe --defaults-file=***file_name* **--port=***port_num*

The **mysqld_safe** script is written so that it normally can start a server that was installed from either a source or a binary distribution of MySQL, even though these types of distributions typically install the server in slightly different locations. (See Section 2.1.4, "Installation Layouts".) **mysqld_safe** expects one of the following conditions to be true:

- The server and databases can be found relative to the working directory (the directory from which **mysqld_safe** is invoked). For binary distributions, **mysqld_safe** looks under its working directory for bin and data directories. For source distributions, it looks for libexec and var directories. This condition should be met if you execute **mysqld_safe** from your MySQL installation directory (for example, /usr/local/mysql for a binary distribution).
- If the server and databases cannot be found relative to the working directory, **mysqld_safe** attempts to locate them by absolute path names. Typical locations are /usr/local/libexec and /usr/local/var. The actual locations are determined from the values configured into the distribution at the time it was built. They should be correct if MySQL is installed in the location specified at configuration time.

Because **mysqld_safe** tries to find the server and databases relative to its own working directory, you can install a binary distribution of MySQL anywhere, as long as you run **mysqld_safe** from the MySQL installation directory:

shell> **cd** *mysql_installation_directory*
shell> **bin/mysqld_safe &**

If **mysqld_safe** fails, even when invoked from the MySQL installation directory, specify the **--ledir** and **--datadir** options to indicate the directories in which the server and databases are located on your system.

Beginning with MySQL 5.5.21, **mysqld_safe** tries to use the **sleep** and **date** system utilities to determine how many times it has attempted to start this second, and—if these are present and this is greater than 5 times—is forced to wait 1 full second before starting again. This is intended to prevent excessive CPU usage in the event of repeated failures. (Bug #11761530, Bug #54035)

When you use **mysqld_safe** to start **mysqld**, **mysqld_safe** arranges for error (and notice) messages from itself and from **mysqld** to go to the same destination.

There are several **mysqld_safe** options for controlling the destination of these messages:

- **--log-error=***file_name*: Write error messages to the named error file.
- **--syslog**: Write error messages to syslog on systems that support the **logger** program.
- **--skip-syslog**: Do not write error messages to syslog. Messages are written to the default error log file (*host_name*.err in the data directory), or to a named file if the **--log-error** option is given.

If none of these options is given, the default is **--skip-syslog**.

If **--log-error** and **--syslog** are both given, a warning is issued and **--log-error** takes precedence.

When **mysqld_safe** writes a message, notices go to the logging destination (syslog or the error log file) and stdout. Errors go to the logging destination and stderr.

Normally, you should not edit the **mysqld_safe** script. Instead, configure **mysqld_safe** by using command-line options or options in the [mysqld_safe] section of a my.cnf option file. In rare cases, it might be necessary to edit **mysqld_safe** to get it to start the server properly. However, if you do this, your modified version of **mysqld_safe** might be overwritten if you upgrade MySQL in the future, so you should make a copy of your edited version that you can reinstall.

COPYRIGHT

Copyright 1997, 2018, Oracle and/or its affiliates. All rights reserved.

This documentation is free software; you can redistribute it and/or modify it only under the terms of the GNU General Public License as published by the Free Software Foundation; version 2 of the License.

This documentation is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with the program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA or see 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/).