



Full credit is given to the above companies including the OS that this PDF file was generated!

Rocky Enterprise Linux 9.2 Manual Pages on command 'mysqlslap.1'

\$ man mysqlslap.1

MYSQLSLAP(1) MySQL Database System MYSQLSLAP(1)

NAME

mysqlslap - a load emulation client

SYNOPSIS

mysqlslap [options]

DESCRIPTION

mysqlslap is a diagnostic program designed to emulate client load for a MySQL server and to report the timing of each stage. It works as if multiple clients are accessing the server.

Invoke mysqlslap like this:

mysqlslap [options]

Some options such as --create or --query enable you to specify a string containing an SQL statement or a file containing statements. If you specify a file, by default it must contain one statement per line.

(That is, the implicit statement delimiter is the newline character.)

Use the --delimiter option to specify a different delimiter, which enables you to specify statements that span multiple lines or place multiple statements on a single line. You cannot include comments in a

file; mysqlslap does not understand them.

mysqlslap runs in three stages:

1. Create schema, table, and optionally any stored programs or data to use for the test. This stage uses a single client connection.
2. Run the load test. This stage can use many client connections.
3. Clean up (disconnect, drop table if specified). This stage uses a single client connection.

Examples:

Supply your own create and query SQL statements, with 50 clients querying and 200 selects for each (enter the command on a single line):

```
mysqlslap --delimiter=";"  
--create="CREATE TABLE a (b int);INSERT INTO a VALUES (23)"  
--query="SELECT * FROM a" --concurrency=50 --iterations=200
```

Let mysqlslap build the query SQL statement with a table of two INT columns and three VARCHAR columns. Use five clients querying 20 times each. Do not create the table or insert the data (that is, use the previous test's schema and data):

```
mysqlslap --concurrency=5 --iterations=20  
--number-int-cols=2 --number-char-cols=3  
--auto-generate-sql
```

Tell the program to load the create, insert, and query SQL statements from the specified files, where the create.sql file has multiple table creation statements delimited by ';' and multiple insert statements delimited by ';'. The --query file should contain multiple queries delimited by ';'. Run all the load statements, then run all the queries in the query file with five clients (five times each):

```
mysqlslap --concurrency=5  
--iterations=5 --query=query.sql --create=create.sql  
--delimiter=";"
```

mysqlslap supports the following options, which can be specified on the command line or in the [mysqlslap] and [client] groups of an option file. For information about option files used by MySQL programs, see Section 4.2.2.2, "Using Option Files".

- ? --help, -? Display a help message and exit.
- ? --auto-generate-sql, -a Generate SQL statements automatically when they are not supplied in files or using command options.
- ? --auto-generate-sql-add-autoincrement Add an AUTO_INCREMENT column to automatically generated tables.
- ? --auto-generate-sql-execute-number=N Specify how many queries to generate automatically.
- ? --auto-generate-sql-guid-primary Add a GUID-based primary key to automatically generated tables.
- ? --auto-generate-sql-load-type=type Specify the test load type. The permissible values are read (scan tables), write (insert into tables), key (read primary keys), update (update primary keys), or mixed (half inserts, half scanning selects). The default is mixed.
- ? --auto-generate-sql-secondary-indexes=N Specify how many secondary indexes to add to automatically generated tables. By default, none are added.
- ? --auto-generate-sql-unique-query-number=N How many different queries to generate for automatic tests. For example, if you run a key test that performs 1000 selects, you can use this option with a value of 1000 to run 1000 unique queries, or with a value of 50 to perform 50 different selects. The default is 10.
- ? --auto-generate-sql-unique-write-number=N How many different queries to generate for --auto-generate-sql-write-number. The default is 10.
- ? --auto-generate-sql-write-number=N How many row inserts to perform. The default is 100.
- ? --commit=N How many statements to execute before committing. The default is 0 (no commits are done).
- ? --compress, -C Compress all information sent between the client and the server if possible. See Section 4.2.8, ?Connection Compression Control?.

As of MySQL 8.0.18, this option is deprecated. Expect it to be removed in a future version of MySQL. See the section called

?Configuring Legacy Connection Compression?.

? --compression-algorithms=value The permitted compression algorithms for connections to the server. The available algorithms are the same as for the protocol_compression_algorithms system variable.

The default value is uncompressed.

For more information, see Section 4.2.8, ?Connection Compression Control?.

This option was added in MySQL 8.0.18.

? --concurrency=N, -c N The number of parallel clients to simulate.

? --create=value The file or string containing the statement to use for creating the table.

? --create-schema=value The schema in which to run the tests.

Note

If the --auto-generate-sql option is also given, mysqlslap drops the schema at the end of the test run. To avoid this, use the --no-drop option as well.

? --csv[=file_name] Generate output in comma-separated values format.

The output goes to the named file, or to the standard output if no file is given.

? --debug[=debug_options], -# [debug_options] Write a debugging log.

A typical debug_options string is d:t:o,file_name. The default is d:t:o,/tmp/mysqlslap.trace.

This option is available only if MySQL was built using WITH_DEBUG.

MySQL release binaries provided by Oracle are not built using this option.

? --debug-check Print some debugging information when the program exits.

This option is available only if MySQL was built using WITH_DEBUG.

MySQL release binaries provided by Oracle are not built using this option.

? --debug-info, -T Print debugging information and memory and CPU

usage statistics when the program exits.

This option is available only if MySQL was built using WITH_DEBUG.

MySQL release binaries provided by Oracle are not built using this option.

? `--default-auth=plugin` A hint about which client-side authentication plugin to use. See Section 6.2.17, [?Pluggable Authentication?](#).

? `--defaults-extra-file=file_name` Read this option file after the global option file but (on Unix) before the user option file. If the file does not exist or is otherwise inaccessible, an error occurs. If `file_name` is not an absolute path name, it is interpreted relative to the current directory.

For additional information about this and other option-file options, see Section 4.2.2.3, [?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, an error occurs. If `file_name` is not an absolute path name, it is interpreted relative to the current directory.

Exception: Even with `--defaults-file`, client programs read `.mylogin.cnf`.

For additional information about this and other option-file options, see Section 4.2.2.3, [?Command-Line Options that Affect Option-File Handling?](#).

? `--defaults-group-suffix=str` Read not only the usual option groups, but also groups with the usual names and a suffix of `str`. For example, `mysqlslap` normally reads the `[client]` and `[mysqlslap]` groups. If this option is given as `--defaults-group-suffix=_other`, `mysqlslap` also reads the `[client_other]` and `[mysqlslap_other]` groups.

For additional information about this and other option-file options, see Section 4.2.2.3, [?Command-Line Options that Affect Option-File Handling?](#).

? `--delimiter=str, -F str` The delimiter to use in SQL statements supplied in files or using command options.

? `--detach=N` Detach (close and reopen) each connection after each N

statements. The default is 0 (connections are not detached).

? `--enable-cleartext-plugin` Enable the `mysql_clear_password` cleartext authentication plugin. (See Section 6.4.1.4, [?Client-Side Cleartext Pluggable Authentication?](#).)

? `--engine=engine_name, -e engine_name` The storage engine to use for creating tables.

? `--get-server-public-key` Request from the server the RSA public key that it uses for key pair-based password exchange. This option applies to clients that connect to the server using an account that authenticates with the `caching_sha2_password` authentication plugin. For connections by such accounts, the server does not send the public key to the client unless requested. The option is ignored for accounts that do not authenticate with that plugin. It is also ignored if RSA-based password exchange is not needed, as is the case when the client connects to the server using a secure connection.

If `--server-public-key-path=file_name` is given and specifies a valid public key file, it takes precedence over `--get-server-public-key`.

For information about the `caching_sha2_password` plugin, see Section 6.4.1.2, [?Caching SHA-2 Pluggable Authentication?](#).

? `--host=host_name, -h host_name` Connect to the MySQL server on the given host.

? `--iterations=N, -i N` The number of times to run the tests.

? `--login-path=name` Read options from the named login path in the `.mylogin.cnf` login path file. A [?login path?](#) is an option group containing options that specify which MySQL server to connect to and which account to authenticate as. To create or modify a login path file, use the `mysql_config_editor` utility. See [mysql_config_editor\(1\)](#).

For additional information about this and other option-file options, see Section 4.2.2.3, [?Command-Line Options that Affect Option-File Handling?](#).

- ? --no-drop Prevent mysqlslap from dropping any schema it creates during the test run.
- ? --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. The exception is that the .mylogin.cnf file is read in all cases, if it exists. This permits passwords to be specified in a safer way than on the command line even when --no-defaults is used. To create .mylogin.cnf, use the mysql_config_editor utility. See mysql_config_editor(1). For additional information about this and other option-file options, see Section 4.2.2.3, "Command-Line Options that Affect Option-File Handling".
- ? --number-char-cols=N, -x N The number of VARCHAR columns to use if --auto-generate-sql is specified.
- ? --number-int-cols=N, -y N The number of INT columns to use if --auto-generate-sql is specified.
- ? --number-of-queries=N Limit each client to approximately this many queries. Query counting takes into account the statement delimiter. For example, if you invoke mysqlslap as follows, the ; delimiter is recognized so that each instance of the query string counts as two queries. As a result, 5 rows (not 10) are inserted.
- ```
mysqlslap --delimiter=";" --number-of-queries=10
 --query="use test;insert into t values(null)"
```
- ? --only-print Do not connect to databases. mysqlslap only prints what it would have done.
- ? --password[=password], -p[password] The password of the MySQL account used for connecting to the server. The password value is optional. If not given, mysqlslap prompts for one. If given, there must be no space between --password= or -p and the password following it. If no password option is specified, the default is to send no password.
- Specifying a password on the command line should be considered

insecure. To avoid giving the password on the command line, use an option file. See Section 6.1.2.1, "End-User Guidelines for Password Security".

To explicitly specify that there is no password and that mysqlslap should not prompt for one, use the `--skip-password` option.

? `--password1[=pass_val]` The password for multifactor authentication factor 1 of the MySQL account used for connecting to the server.

The password value is optional. If not given, mysqlslap prompts for one. If given, there must be no space between `--password1=` and the password following it. If no password option is specified, the default is to send no password.

Specifying a password on the command line should be considered insecure. To avoid giving the password on the command line, use an option file. See Section 6.1.2.1, "End-User Guidelines for Password Security".

To explicitly specify that there is no password and that mysqlslap should not prompt for one, use the `--skip-password1` option.

`--password1` and `--password` are synonymous, as are `--skip-password1` and `--skip-password`.

? `--password2[=pass_val]` The password for multifactor authentication factor 2 of the MySQL account used for connecting to the server.

The semantics of this option are similar to the semantics for `--password1`; see the description of that option for details.

? `--password3[=pass_val]` The password for multifactor authentication factor 3 of the MySQL account used for connecting to the server.

The semantics of this option are similar to the semantics for `--password1`; see the description of that option for details.

? `--pipe, -W` On Windows, connect to the server using a named pipe.

This option applies only if the server was started with the `named_pipe` system variable enabled to support named-pipe connections. In addition, the user making the connection must be a member of the Windows group specified by the `named_pipe_full_access_group` system variable.



- ? `--plugin-dir=dir_name` The directory in which to look for plugins. Specify this option if the `--default-auth` option is used to specify an authentication plugin but `mysqlslap` does not find it. See Section 6.2.17, `?Pluggable Authentication?`.
- ? `--port=port_num, -P port_num` For TCP/IP connections, the port number to use.
- ? `--post-query=value` The file or string containing the statement to execute after the tests have completed. This execution is not counted for timing purposes.
- ? `--post-system=str` The string to execute using `system()` after the tests have completed. This execution is not counted for timing purposes.
- ? `--pre-query=value` The file or string containing the statement to execute before running the tests. This execution is not counted for timing purposes.
- ? `--pre-system=str` The string to execute using `system()` before running the tests. This execution is not counted for timing purposes.
- ? `--print-defaults` Print the program name and all options that it gets from option files.  
  
For additional information about this and other option-file options, see Section 4.2.2.3, `?Command-Line Options that Affect Option-File Handling?`.
- ? `--protocol={TCP|SOCKET|PIPE|MEMORY}` The transport protocol to use for connecting to the server. It is useful when the other connection parameters normally result in use of a protocol other than the one you want. For details on the permissible values, see Section 4.2.7, `?Connection Transport Protocols?`.
- ? `--query=value, -q value` The file or string containing the SELECT statement to use for retrieving data.
- ? `--server-public-key-path=file_name` The path name to a file in PEM format containing a client-side copy of the public key required by the server for RSA key pair-based password exchange. This option

applies to clients that authenticate with the sha256\_password or caching\_sha2\_password authentication plugin. This option is ignored for accounts that do not authenticate with one of those plugins. It is also ignored if RSA-based password exchange is not used, as is the case when the client connects to the server using a secure connection.

If --server-public-key-path=file\_name is given and specifies a valid public key file, it takes precedence over --get-server-public-key.

For sha256\_password, this option applies only if MySQL was built using OpenSSL.

For information about the sha256\_password and caching\_sha2\_password plugins, see Section 6.4.1.3, "SHA-256 Pluggable Authentication", and Section 6.4.1.2, "Caching SHA-2 Pluggable Authentication".

? --shared-memory-base-name=name On Windows, the shared-memory name to use for connections made using shared memory to a local server. The default value is MYSQL. The shared-memory name is case-sensitive.

This option applies only if the server was started with the shared\_memory system variable enabled to support shared-memory connections.

? --silent, -s Silent mode. No output.

? --socket=path, -S path For connections to localhost, the Unix socket file to use, or, on Windows, the name of the named pipe to use.

On Windows, this option applies only if the server was started with the named\_pipe system variable enabled to support named-pipe connections. In addition, the user making the connection must be a member of the Windows group specified by the named\_pipe\_full\_access\_group system variable.

? --sql-mode=mode Set the SQL mode for the client session.

? --ssl\* Options that begin with --ssl specify whether to connect to the server using encryption and indicate where to find SSL keys and

certificates. See the section called "Command Options for Encrypted Connections".

? `--ssl-fips-mode={OFF|ON|STRICT}` Controls whether to enable FIPS mode on the client side. The `--ssl-fips-mode` option differs from other `--ssl-xxx` options in that it is not used to establish encrypted connections, but rather to affect which cryptographic operations to permit. See Section 6.8, "FIPS Support".

These `--ssl-fips-mode` values are permitted:

? OFF: Disable FIPS mode.

? ON: Enable FIPS mode.

? STRICT: Enable "strict" FIPS mode.

#### Note

If the OpenSSL FIPS Object Module is not available, the only permitted value for `--ssl-fips-mode` is OFF. In this case, setting `--ssl-fips-mode` to ON or STRICT causes the client to produce a warning at startup and to operate in non-FIPS mode.

As of MySQL 8.0.34, this option is deprecated. Expect it to be removed in a future version of MySQL.

? `--tls-ciphersuites=ciphersuite_list` The permissible ciphersuites for encrypted connections that use TLSv1.3. The value is a list of one or more colon-separated ciphersuite names. The ciphersuites that can be named for this option depend on the SSL library used to compile MySQL. For details, see Section 6.3.2, "Encrypted Connection TLS Protocols and Ciphers".

This option was added in MySQL 8.0.16.

? `--tls-version=protocol_list` The permissible TLS protocols for encrypted connections. The value is a list of one or more comma-separated protocol names. The protocols that can be named for this option depend on the SSL library used to compile MySQL. For details, see Section 6.3.2, "Encrypted Connection TLS Protocols and Ciphers".

? `--user=user_name, -u user_name` The user name of the MySQL account to use for connecting to the server.

? --verbose, -v Verbose mode. Print more information about what the program does. This option can be used multiple times to increase the amount of information.

? --version, -V Display version information and exit.

? --zstd-compression-level=level The compression level to use for connections to the server that use the zstd compression algorithm. The permitted levels are from 1 to 22, with larger values indicating increasing levels of compression. The default zstd compression level is 3. The compression level setting has no effect on connections that do not use zstd compression.

For more information, see Section 4.2.8, "Connection Compression Control".

This option was added in MySQL 8.0.18.

## COPYRIGHT

Copyright ? 1997, 2023, Oracle and/or its affiliates.

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/>).