



**Full credit is given to the above companies including the Operating System (OS) that this PDF file was generated!**

***Rocky Enterprise Linux 9.2 Manual Pages on command 'mysqladmin.1'***

**\$ man mysqladmin.1**

MYSQLADMIN(1)                      MySQL Database System                      MYSQLADMIN(1)

NAME

mysqladmin - a MySQL server administration program

SYNOPSIS

mysqladmin [options] command [command-options] [command [command-options]] ...

DESCRIPTION

mysqladmin is a client for performing administrative operations. You can use it to check the server's configuration and current status, to create and drop databases, and more.

Invoke mysqladmin like this:

mysqladmin [options] command [command-arg] [command [command-arg]] ...

mysqladmin supports the following commands. Some of the commands take an argument following the command name.

? create db\_name

Create a new database named db\_name.

? debug

Prior to MySQL 8.0.20, tell the server to write debug information to the error log.

The connected user must have the SUPER privilege. Format and content of this information is subject to change.

This includes information about the Event Scheduler. See Section 25.4.5, ?Event Scheduler Status?.

? drop db\_name

Delete the database named db\_name and all its tables.

? extended-status

Display the server status variables and their values.

? flush-hosts

Flush all information in the host cache. See Section 5.1.12.3, "DNS Lookups and the Host Cache".

? flush-logs [log\_type ...]

Flush all logs.

The `mysqladmin flush-logs` command permits optional log types to be given, to specify which logs to flush. Following the `flush-logs` command, you can provide a space-separated list of one or more of the following log types: `binary`, `engine`, `error`, `general`, `relay`, `slow`. These correspond to the log types that can be specified for the `FLUSH LOGS` SQL statement.

? flush-privileges

Reload the grant tables (same as `reload`).

? flush-status

Clear status variables.

? flush-tables

Flush all tables.

? flush-threads

Flush the thread cache.

? kill id,id,...

Kill server threads. If multiple thread ID values are given, there must be no spaces in the list.

To kill threads belonging to other users, the connected user must have the `CONNECTION_ADMIN` privilege (or the deprecated `SUPER` privilege).

? password new\_password

Set a new password. This changes the password to `new_password` for the account that you use with `mysqladmin` for connecting to the server. Thus, the next time you invoke `mysqladmin` (or any other client program) using the same account, you must specify the new password.

Warning

Setting a password using `mysqladmin` should be considered insecure. On some systems, your password becomes visible to system status programs such as `ps` that may be invoked by other users to display command lines. MySQL clients typically

overwrite the command-line password argument with zeros during their initialization sequence. However, there is still a brief interval during which the value is visible. Also, on some systems this overwriting strategy is ineffective and the password remains visible to ps. (SystemV Unix systems and perhaps others are subject to this problem.)

If the `new_password` value contains spaces or other characters that are special to your command interpreter, you need to enclose it within quotation marks. On Windows, be sure to use double quotation marks rather than single quotation marks; single quotation marks are not stripped from the password, but rather are interpreted as part of the password. For example:

```
mysqladmin password "my new password"
```

The new password can be omitted following the password command. In this case, `mysqladmin` prompts for the password value, which enables you to avoid specifying the password on the command line. Omitting the password value should be done only if password is the final command on the `mysqladmin` command line. Otherwise, the next argument is taken as the password.

#### Caution

Do not use this command used if the server was started with the `--skip-grant-tables` option. No password change is applied. This is true even if you precede the password command with `flush-privileges` on the same command line to re-enable the grant tables because the flush operation occurs after you connect. However, you can use `mysqladmin flush-privileges` to re-enable the grant table and then use a separate `mysqladmin password` command to change the password.

#### ? ping

Check whether the server is available. The return status from `mysqladmin` is 0 if the server is running, 1 if it is not. This is 0 even in case of an error such as Access denied, because this means that the server is running but refused the connection, which is different from the server not running.

#### ? processlist

Show a list of active server threads. This is like the output of the `SHOW PROCESSLIST` statement. If the `--verbose` option is given, the output is like that of `SHOW FULL PROCESSLIST`. (See Section 13.7.7.29, `?SHOW PROCESSLIST Statement?.`)

#### ? reload

Reload the grant tables.

? refresh

Flush all tables and close and open log files.

? shutdown

Stop the server.

? start-replica

Start replication on a replica server. Use this command from MySQL 8.0.26.

? start-slave

Start replication on a replica server. Use this command before MySQL 8.0.26.

? status

Display a short server status message.

? stop-replica

Stop replication on a replica server. Use this command from MySQL 8.0.26.

? stop-slave

Stop replication on a replica server. Use this command before MySQL 8.0.26.

? variables

Display the server system variables and their values.

? version

Display version information from the server.

All commands can be shortened to any unique prefix. For example:

```
$> mysqladmin proc stat
```

```
+----+-----+-----+----+-----+-----+-----+-----+
| Id | User | Host   | db | Command | Time | State | Info          |
+----+-----+-----+----+-----+-----+-----+-----+
| 51 | jones | localhost |   | Query   | 0    |      | show processlist |
```

```
Uptime: 1473624 Threads: 1 Questions: 39487
```

```
Slow queries: 0 Opens: 541 Flush tables: 1
```

```
Open tables: 19 Queries per second avg: 0.0268
```

The mysqladmin status command result displays the following values:

? Uptime

The number of seconds the MySQL server has been running.

? Threads

The number of active threads (clients).

? Questions

The number of questions (queries) from clients since the server was started.

? Slow queries

The number of queries that have taken more than long\_query\_time seconds. See Section 5.4.5, "The Slow Query Log".

? Opens

The number of tables the server has opened.

? Flush tables

The number of flush-\*, refresh, and reload commands the server has executed.

? Open tables

The number of tables that currently are open.

If you execute mysqladmin shutdown when connecting to a local server using a Unix socket file, mysqladmin waits until the server's process ID file has been removed, to ensure that the server has stopped properly.

mysqladmin supports the following options, which can be specified on the command line or in the [mysqladmin] 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, -?

??

?Command-Line Format ? --help ?

??

Display a help message and exit.

? --bind-address=ip\_address

??

?Command-Line Format ? --bind-address=ip\_address ?

??

On a computer having multiple network interfaces, use this option to select which interface to use for connecting to the MySQL server.

? --character-sets-dir=dir\_name

??

?Command-Line Format ? --character-sets-dir=path ?

??

?Type ? String ?  
??  
?Default Value ? [none] ?  
??

The directory where character sets are installed. See Section 10.15, ?Character Set Configuration?.

? --compress, -C  
??  
?Command-Line Format ? --compress[={OFF|ON}] ?  
??

?Deprecated ? 8.0.18 ?  
??  
?Type ? Boolean ?  
??  
?Default Value ? OFF ?  
??

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  
??  
?Command-Line Format ? --compression-algorithms=value ?  
??  
?Introduced ? 8.0.18 ?  
??  
?Type ? Set ?  
??  
?Default Value ? uncompressed ?  
??  
?Valid Values ? ?  
? ? zlib ?  
? ? ?

? ? zstd ?  
? ? ? ?  
? ? uncompressed ?

??

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.

? --connect-timeout=value

??

?Command-Line Format ? --connect-timeout=value ?

??

?Type ? Numeric ?

??

?Default Value ? 43200 ?

??

The maximum number of seconds before connection timeout. The default value is 43200 (12 hours).

? --count=N, -c N

??

?Command-Line Format ? --count=# ?

??

The number of iterations to make for repeated command execution if the --sleep option is given.

? --debug[=debug\_options], -# [debug\_options]

??

?Command-Line Format ? --debug[=debug\_options] ?

??

?Type ? String ?

??

?Default Value ? d:t:o,/tmp/mysqladmin.trace ?

??

Write a debugging log. A typical debug\_options string is d:t:o,file\_name. The default is d:t:o,/tmp/mysqladmin.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

??

?Command-Line Format ? --debug-check ?

??

?Type ? Boolean ?

??

?Default Value ? FALSE ?

??

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

??

?Command-Line Format ? --debug-info ?

??

?Type ? Boolean ?

??

?Default Value ? FALSE ?

??

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

??

?Command-Line Format ? --default-auth=plugin ?

??

?Type ? String ?

??



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

? --default-character-set=charset\_name

??

?Command-Line Format ? --default-character- ?

? set=charset\_name ?

??

?Type ? String ?

??

Use charset\_name as the default character set. See Section 10.15, [?Character Set Configuration?](#).

? --defaults-extra-file=file\_name

??

?Command-Line Format ? --defaults-extra-file=file\_name ?

??

?Type ? 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

??

?Command-Line Format ? --defaults-file=file\_name ?

??

?Type ? 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

??

?Command-Line Format ? --defaults-group-suffix=str ?

??

?Type ? String ?

??

Read not only the usual option groups, but also groups with the usual names and a suffix of str. For example, mysqladmin normally reads the [client] and [mysqladmin] groups. If this option is given as --defaults-group-suffix=\_other, mysqladmin also reads the [client\_other] and [mysqladmin\_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".

? --enable-cleartext-plugin

??

?Command-Line Format ? --enable-cleartext-plugin ?

??

?Type ? Boolean ?

??

?Default Value ? FALSE ?

??

Enable the mysql\_clear\_password cleartext authentication plugin. (See Section 6.4.1.4, "Client-Side Cleartext Pluggable Authentication".)

? --force, -f

??

?Command-Line Format ? --force ?

??

Do not ask for confirmation for the drop db\_name command. With multiple commands, continue even if an error occurs.

? --get-server-public-key

??

?Command-Line Format ? --get-server-public-key ?

??

?Type ? Boolean ?

??

Request from the server the public key required for RSA key pair-based password exchange. This option applies to clients that authenticate with the caching\_sha2\_password authentication plugin. For that plugin, the server does not send the public key unless requested. This option is ignored for accounts that do not authenticate with that plugin. 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 information about the caching\_sha2\_password plugin, see Section 6.4.1.2, "Caching SHA-2 Pluggable Authentication".

? --host=host\_name, -h host\_name

??

?Command-Line Format ? --host=host\_name ?

??

?Type ? String ?

??

?Default Value ? localhost ?

??

Connect to the MySQL server on the given host.

? --login-path=name

??

?Command-Line Format ? --login-path=name ?

??

?Type ? String ?

??

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-beep, -b

??

?Command-Line Format ? --no-beep ?

??

Suppress the warning beep that is emitted by default for errors such as a failure to connect to the server.

? --no-defaults

??

?Command-Line Format ? --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".

? --password[=password], -p[password]

??

?Command-Line Format ? --password[=password] ?

??

?Type           ? String           ?

??

The password of the MySQL account used for connecting to the server. The password value is optional. If not given, mysqladmin 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 mysqladmin 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, mysql 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 mysqladmin 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

????????????????????????????????????

?Command-Line Format ? --pipe ?

????????????????????????????????????

?Type ? String ?

????????????????????????????????????

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

??

?Command-Line Format ? --plugin-dir=dir\_name ?

??

?Type ? Directory 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 mysqladmin does not find it. See Section 6.2.17, ?Pluggable Authentication?.

? --port=port\_num, -P port\_num

??

?Command-Line Format ? --port=port\_num ?

??

?Type ? Numeric ?

??

?Default Value ? 3306 ?

??

For TCP/IP connections, the port number to use.

? --print-defaults

??

?Command-Line Format ? --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}

??

?Command-Line Format ? --protocol=type ?

??

?Type ? String ?

??

?Default Value ? [see text] ?

??

?Valid Values ? ?

? ? 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".

? --relative, -r

??

?Command-Line Format ? --relative ?

??

Show the difference between the current and previous values when used with the --sleep option. This option works only with the extended-status command.

? --server-public-key-path=file\_name

??

?Command-Line Format ? --server-public-key- ?

? ? path=file\_name ?

??

?Type ? 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

??

?Command-Line Format ? --shared-memory-base-name=name ?

??

?Platform Specific ? Windows ?

??

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.

? --show-warnings

??

?Command-Line Format ? --show-warnings ?

??

Show warnings resulting from execution of statements sent to the server.

? --shutdown-timeout=value

??

?Command-Line Format ? --shutdown-timeout=seconds ?

??

?Type ? Numeric ?

??

?Default Value ? 3600 ?

??

The maximum number of seconds to wait for server shutdown. The default value is 3600 (1 hour).

? --silent, -s

??

?Command-Line Format ? --silent ?

??



Exit silently if a connection to the server cannot be established.

? --sleep=delay, -i delay

??

?Command-Line Format ? --sleep=delay ?

??

Execute commands repeatedly, sleeping for delay seconds in between. The --count option determines the number of iterations. If --count is not given, mysqladmin executes commands indefinitely until interrupted.

? --socket=path, -S path

??

?Command-Line Format ? --socket={file\_name|pipe\_name} ?

??

?Type ? String ?

??

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.

? --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}

??

?Command-Line Format ? --ssl-fips-mode={OFF|ON|STRICT} ?

??

?Deprecated ? 8.0.34 ?

??

?Type ? Enumeration ?

??

?Default Value ? OFF ?

??

?Valid Values ? ?

? ? 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

??

?Command-Line Format ? --tls- ?

? ? ciphersuites=ciphersuite\_list ?

??

?Introduced ? 8.0.16 ?

??

?Type ? String ?

??

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,



?Command-Line Format ? --verbose ?

??

Verbose mode. Print more information about what the program does.

? --version, -V

??

?Command-Line Format ? --version ?

??

Display version information and exit.

? --vertical, -E

??

?Command-Line Format ? --vertical ?

??

Print output vertically. This is similar to --relative, but prints output vertically.

? --wait[=count], -w[count]

??

?Command-Line Format ? --wait ?

??

If the connection cannot be established, wait and retry instead of aborting. If a count value is given, it indicates the number of times to retry. The default is one time.

? --zstd-compression-level=level

??

?Command-Line Format ? --zstd-compression-level=# ?

??

?Introduced ? 8.0.18 ?

??

?Type ? Integer ?

??

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/>).

MySQL 8.0

11/27/2023

MYSQLADMIN(1)