

FULL-SLOW-LOG

Collect MySQL queries in a distinct slow-query log file, for a specific period of time.

This tool supports enhanced log features in Percona Server 5.1 and MariaDB 5.1.

Usage

```
$ full-slow-log [ -v ] [ -s seconds ] [ -f file ] [ -c config ] [ -l time ]
```

Command Options

-v | --verbose

Enable verbose output.

-s | --sleep-time *seconds*

Specify the number of seconds to sleep while logs are being written to the log file. The default is 5 seconds, which is probably too short for most sites, but the value is chosen to be as low impact as possible if you forget to give another value.

-f | --file *file*

Specify a slow-query log destination. The default is to create a new file based on the current slow-query log file's name and destination, appending to the name a suffix with datetime information: "-full-YYYYMMDDHHMMSS".

-c | --config-file *file*

Specify a MySQL config file. You can store host, user, and password in the [client] section of the config file. The default is \$HOME/.my.cnf, like other mysql client programs.

-l | --long-query-time *seconds*

Specify the threshold for logging queries to the slow-query log. The default is full logging, i.e. long-query-time=0.

Basic Log Control

This tool also uses environment variables as an alternative way to control the tool behavior. Command-line options take priority over environment variables.

For example, the following commands would make the script collect query logs for 300 seconds and 60 seconds, respectively.

```
$ export GETSLOW_SLEEP_TIME=300
$ full-slow-log
$ full-slow-log -s 60
```

GETSLOW_SLEEP_TIME

Works like the --sleep-time command-line option.

GETSLOW_LOG_FILE

Works like the --file command-line option.

GETSLOW_CONFIG_FILE

Works like the --config-file command-line option.

GETSLOW_LONG_QUERY_TIME

Works like the --long-query-time command-line option.

Enhanced Log Control

Percona Server and MariaDB support additional configuration for the slow-query log file. You can control these features with environment variables:

The following environment variables apply logging changes to both Percona Server and MariaDB:

GETSLOW_VERBOSITY

Add more detail to the slow-query log. The value is a comma-separated list of keywords: `microtime`, `query_plan`, `innodb`, `profiling`, `profiling_getrusage`. The default includes the first four keywords. The `profiling` and `profiling_getrusage` keywords are supported only by Percona Server 5.5

GETSLOW_LOG_FILTER

Log only queries that invoke specific data access behavior. The value is a comma-separated string of keywords: `qc_miss`, `full_scan`, `full_join`, `tmp_table`, `tmp_table_on_disk`, `filesort`, `filesort_on_disk`. The default is no change to the configured log filter.

GETSLOW_RATE_LIMIT

Specify an integer *N*. One out of *N* sessions or queries are logged, the rest are not logged. The default is no change to the configured log rate limit.

The following environment variables apply logging changes only to Percona Server:

GETSLOW_RATE_TYPE

Specify `session` or `query` to rate-limit the log entries by query or by session. The default is no change to the configured log rate type. This is supported in Percona Server 5.5 only.

GETSLOW_SP_STATEMENTS

Set to 1 to log statements run from stored routines.

GETSLOW_TIMESTAMP_ALWAYS

Set to 1 to make every slow-query log record include a timestamp. The default is no change to the configured log format. Note this variable's name is different in Percona Server 5.1 versus 5.5, but you can use the environment variable as named here, and the tool will adjust for 5.1.

GETSLOW_TIMESTAMP_PRECISION

Set to `second` or `microsecond`. The default is no change to the configured log format.

The following enhanced log control features are not supported by this tool:

Log slave statements

Enabling or disabling this feature requires stopping and starting the slave threads. It could cause interruptions on the slave if this tool were to stop the slave threads. You can always enable this option on Percona Server manually, but not using this tool.

Affecting Long-Running Sessions

Changing global values dynamically does not affect current MySQL sessions. Only new sessions new sessions that start after the global value has been changed heed the change. This means that applications with long-running sessions, or connection pools, etc. might miss the temporary changes made by this tool.

Percona Server has a configuration variable `slow_query_log_use_global_control`. This forces running threads to heed global changes dynamically. However, changing this variable is itself a global change, which the running threads won't heed by default.

You can change this variable manually, but not using this tool. Then restart any long-running sessions or connection pools, and subsequently they should heed dynamic changes to global variables.