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

$ man mysqlrepair.1

MYSQLCHECK(1)   MySQL Database System   MYSQLCHECK(1)

NAME

    mysqlcheck - a table maintenance program

SYNOPSIS

    mysqlcheck [options] [db_name [tbl_name ...]]

DESCRIPTION

    The mysqlcheck client performs table maintenance: It checks, repairs, optimizes, or

    analyzes tables.

    Each table is locked and therefore unavailable to other sessions while it is being

    processed, although for check operations, the table is locked with a READ lock only (see

    Section 13.3.6, ?LOCK TABLES and UNLOCK TABLES Statements?, for more information about

    READ and WRITE locks). Table maintenance operations can be time-consuming, particularly

    for large tables. If you use the --databases or --all-databases option to process all

    tables in one or more databases, an invocation of mysqlcheck might take a long time. (This

    is also true for the MySQL upgrade procedure if it determines that table checking is

    needed because it processes tables the same way.)

    mysqlcheck must be used when the mysqld server is running, which means that you do not

    have to stop the server to perform table maintenance.

    mysqlcheck uses the SQL statements CHECK TABLE, REPAIR TABLE, ANALYZE TABLE, and OPTIMIZE

    TABLE in a convenient way for the user. It determines which statements to use for the

    operation you want to perform, and then sends the statements to the server to be executed.

    For details about which storage engines each statement works with, see the descriptions

    for those statements in Section 13.7.3, ?Table Maintenance Statements?.

All storage engines do not necessarily support all four maintenance operations. In such cases, an error message is displayed. For example, if test.t is an MEMORY table, an attempt to check it produces this result:

```
$> mysqlcheck test t
test.t
note     : The storage engine for the table doesn't support check
```

If mysqlcheck is unable to repair a table, see Section 2.10.13, ?Rebuilding or Repairing Tables or Indexes? for manual table repair strategies. This is the case, for example, for InnoDB tables, which can be checked with CHECK TABLE, but not repaired with REPAIR TABLE.

Caution

It is best to make a backup of a table before performing a table repair operation; under some circumstances the operation might cause data loss. Possible causes include but are not limited to file system errors.

There are three general ways to invoke mysqlcheck:

```
mysqlcheck [options] db_name [tbl_name ...]
mysqlcheck [options] --databases db_name ...
mysqlcheck [options] --all-databases
```

If you do not name any tables following db_name or if you use the --databases or --all-databases option, entire databases are checked.

mysqlcheck has a special feature compared to other client programs. The default behavior of checking tables (--check) can be changed by renaming the binary. If you want to have a tool that repairs tables by default, you should just make a copy of mysqlcheck named mysqlrepair, or make a symbolic link to mysqlcheck named mysqlrepair. If you invoke mysqlrepair, it repairs tables.

The names shown in the following table can be used to change mysqlcheck default behavior.

??
?Command   ? Meaning   ?
??
?mysqlrepair   ? The default option is --repair   ?
??
?mysqlanalyze ? The default option is --analyze ?
??
?mysqloptimize ? The default option is --optimize ?

??

mysqlcheck supports the following options, which can be specified on the command line or in the [mysqlcheck] 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.

? --all-databases, -A

  ??

  ?Command-Line Format ? --all-databases ?

  ??

  Check all tables in all databases. This is the same as using the --databases option and naming all the databases on the command line, except that the INFORMATION_SCHEMA and performance_schema databases are not checked. They can be checked by explicitly naming them with the --databases option.

? --all-in-1, -1

  ??

  ?Command-Line Format ? --all-in-1 ?

  ??

  Instead of issuing a statement for each table, execute a single statement for each database that names all the tables from that database to be processed.

? --analyze, -a

  ??

  ?Command-Line Format ? --analyze ?

  ??

  Analyze the tables.

? --auto-repair

  ??

  ?Command-Line Format ? --auto-repair ?

  ??

  If a checked table is corrupted, automatically fix it. Any necessary repairs are done

after all tables have been checked.

? --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=dir_name ?

??

?Type   ? Directory name   ?

??

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

Configuration?.

? --check, -c

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

?Command-Line Format ? --check ?

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

Check the tables for errors. This is the default operation.

? --check-only-changed, -C

???

?Command-Line Format ? --check-only-changed ?

???

Check only tables that have changed since the last check or that have not been closed

properly.

? --check-upgrade, -g

???

?Command-Line Format ? --check-upgrade ?

???

Invoke CHECK TABLE with the FOR UPGRADE option to check tables for incompatibilities

with the current version of the server.

? --compress

???

?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.

? --databases, -B

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

?Command-Line Format ? --databases ?

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

Process all tables in the named databases. Normally, mysqlcheck treats the first name

argument on the command line as a database name and any following names as table

names. With this option, it treats all name arguments as database names.

? --debug[=debug_options], -# [debug_options]

??

?Command-Line Format ? --debug[=debug_options] ?

??

?Type   ? String   ?

??

?Default Value   ? d:t:o   ?

??

Write a debugging log. A typical debug_options string is d:t:o,file_name. The default

is d:t:o.

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-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, mysqlcheck normally reads the [client] and [mysqlcheck]

groups. If this option is given as --defaults-group-suffix=_other, mysqlcheck also

reads the [client_other] and [mysqlcheck_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?.

? --extended, -e

???

?Command-Line Format ? --extended ?

???

If you are using this option to check tables, it ensures that they are 100% consistent

but takes a long time.

If you are using this option to repair tables, it runs an extended repair that may not

only take a long time to execute, but may produce a lot of garbage rows also!

? --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?.

? --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?.)

? --fast, -F

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

?Command-Line Format ? --fast ?

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

Check only tables that have not been closed properly.

? --force, -f

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

?Command-Line Format ? --force ?

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

Continue even if an SQL 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?.

? --medium-check, -m

??

?Command-Line Format ? --medium-check ?

??

Do a check that is faster than an --extended operation. This finds only 99.99% of all

errors, which should be good enough in most cases.

? --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?.

? --optimize, -o

???

?Command-Line Format ? --optimize ?

???

Optimize the tables.

? --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, mysqlcheck 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 mysqlcheck 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, mysqlcheck 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 mysqlcheck 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 mysqlcheck 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?.

? --quick, -q

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

?Command-Line Format ? --quick ?

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

If you are using this option to check tables, it prevents the check from scanning the

rows to check for incorrect links. This is the fastest check method.

If you are using this option to repair tables, it tries to repair only the index tree.

This is the fastest repair method.

? --repair, -r

??

?Command-Line Format ? --repair ?

??

Perform a repair that can fix almost anything except unique keys that are not unique.

? --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.

? --silent, -s

?????????????????????????????????????
?Command-Line Format ? --silent ?
?????????????????????????????????????
Silent mode. Print only error messages.

? --skip-database=db_name

???
?Command-Line Format ? --skip-database=db_name ?
???
Do not include the named database (case-sensitive) in the operations performed by mysqlcheck.

- --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.

? --tables

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

?Command-Line Format ? --tables ?

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

Override the --databases or -B option. All name arguments following the option are

regarded as table names.

? --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,

see Section 6.3.2, ?Encrypted Connection TLS Protocols and Ciphers?.

This option was added in MySQL 8.0.16.

- --tls-version=protocol_list

Command-Line Format	--tls-version=protocol_list
Type	String
Default Value (≥ 8.0.16)	TLSv1,TLSv1.1,TLSv1.2,TLSv1.3 (OpenSSL 1.1.1 or higher)
	TLSv1,TLSv1.1,TLSv1.2 (otherwise)
Default Value (≤ 8.0.15)	TLSv1,TLSv1.1,TLSv1.2

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".

- --use-frm

Command-Line Format	--use-frm

For repair operations on MyISAM tables, get the table structure from the data dictionary so that the table can be repaired even if the .MYI header is corrupted.

- --user=user_name, -u user_name

Command-Line Format	--user=user_name,
Type	String

The user name of the MySQL account to use for connecting to the server.

? --verbose, -v

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

?Command-Line Format ? --verbose ?

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

Verbose mode. Print information about the various stages of program operation.

? --version, -V

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

?Command-Line Format ? --version ?

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

Display version information and exit.

? --write-binlog

???

?Command-Line Format ? --write-binlog ?

???

This option is enabled by default, so that ANALYZE TABLE, OPTIMIZE TABLE, and REPAIR

TABLE statements generated by mysqlcheck are written to the binary log. Use

--skip-write-binlog to cause NO_WRITE_TO_BINLOG to be added to the statements so that

they are not logged. Use the --skip-write-binlog when these statements should not be

sent to replicas or run when using the binary logs for recovery from backup.

? --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

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   MYSQLCHECK(1)