MySQL Programs, Invoking MySQL Programs, Specifying Program Options and Option File

Overview of MySQL Programs

- There are many different programs in a MySQL installation.
- Most MySQL distributions include all of these programs, except for those programs that are platform-specific.
- Each MySQL program takes many different options. Most programs provide a --help option that you can use to get a description of the program's different options. For example, try mysql --help.
- You can override default option values for MySQL programs by specifying options on the command line or in an option file.
- **mysqld :** The SQL daemon (that is, the MySQL server). To use client programs, **mysqld** must be running, because clients gain access to databases by connecting to the server
- **mysqld_safe :** A server startup script. mysqld_safe attempts to start mysqld.
- **mysql.server :** A server startup script. This script is used on systems that use System V-style run directories containing scripts that start system services for particular run levels. It invokes mysqld_safe to start the MySQL server.

Overview of MySQL Programs

- **mysql_secure_installation :** This program enables you to improve the security of your MySQL installation.
- **mysql_upgrade :** Prior to MySQL 8.0.16, this program is used after a MySQL upgrade operation. It updates the grant tables with any changes that have been made in newer versions of MySQL, and checks tables for incompatibilities and repairs them if necessary.
- MySQL client programs that connect to the MySQL server:
- **mysql :** The command-line tool for interactively entering SQL statements or executing them from a file in batch mode.
- **mysqadmin :** A client that performs administrative operations, such as creating or dropping databases, reloading the grant tables, flushing tables to disk, and reopening log files. mysqladmin can also be used to retrieve version, process, and status information from the server.
- **mysqlcheck :** A table-maintenance client that checks, repairs, analyzes, and optimizes tables.
- **mysqldump :** A client that dumps a MySQL database into a file as SQL, text, or XML.

Invoking MySQL Programs

- To invoke a MySQL program from the command line (that is, from your shell or command prompt), enter the program name followed by any options or other arguments needed to instruct the program what you want it to do.
- The following commands show some sample program invocations.
- mysql --user=root test
- mysqladmin extended-status variables
- mysqldump -u root personnel
- Arguments that begin with a single or double (-,--) specify program options.
- Nonoption arguments (arguments with no leading dash) provide additional information to the program. For example, the mysql program interprets the first nonoption argument as a database name, so the command mysql --user=root test indicates that you want to use the test database.
- You may find it necessary to invoke MySQL programs using the path name to the bin directory in which they are installed.
- For example, if mysql is installed in /usr/local/mysql/bin, you can run the program by invoking it as mysql, and it is not necessary to invoke it as /usr/local/mysql/bin/mysql.

Specifying Program Options

- There are several ways to specify options for MySQL programs:
- List the options on the command line following the program name. This is common for options that apply to a specific invocation of the program.
- List the options in an option file that the program reads when it starts. This is common for options that you want the program to use each time it runs.
- List the options in environment variables. This method is useful for options that you want to apply each time the program runs.
- Options are processed in order, so if an option is specified multiple times, the last occurrence takes precedence. The following command causes mysql to connect to the server running on localhost:
- mysql -h example.com -h localhost
- MySQL programs determine which options are given first by examining environment variables, then by processing option files, and then by checking the command line.

Specifying Program Options

- Because later options take precedence over earlier ones, the processing order means that environment variables have the lowest precedence and command-line options the highest.
- For the server, one exception applies: The mysqld-auto.cnf option file in the data directory is processed last, so it takes precedence even over command-line options.

Using Option Files

- Most MySQL programs can read startup options from option files (sometimes called configuration files).
- Option files provide a convenient way to specify commonly used options so that they need not be entered on the command line each time you run a program.
- To determine whether a program reads option files, invoke it with the --help option. (For mysqld, use --verbose and --help.) If the program reads option files, the help message indicates which files it looks for and which option groups it recognizes.
- Many option files are plain text files, created using any text editor.
- The programs that read options files include: myisamchk, myisampack, mysql, mysqladmin, mysqlbinlog, mysqlcc, mysqlcheck, mysqld, mysqld_safe, mysqldump, mysqlhotcopy, mysqlimport, mysql.server, and mysqlshow.

Option File Processing Order

File Name	Purpose
/etc/my.cnf	Global options
/etc/mysql/my.cnf	Global options
SYSCONFDIR/my.cnf	Global options
$MYSQL_HOME/my.cnf	Server-specific options (server only)
defaults-extra-file	The file specified with --defaults-extra-file, if any
~/.my.cnf	User-specific options
~/.mylogin.cnf	User-specific login path options (clients only)
DATADIR/mysqld-auto.cnf	System variables persisted with SET PERSIST or SET PERSIST_ONLY (server only)

Using Option Files

- In the preceding table, ~ represents the current user's home directory (the value of $HOME).
- SYSCONFDIR represents the directory specified with the SYSCONFDIR option to CMake when MySQL was built. By default, this is the etc directory located under the compiled-in installation directory.
- MYSQL_HOME is an environment variable containing the path to the directory in which the server-specific my.cnf file resides. If MYSQL_HOME is not set and you start the server using the mysqld_safe program, mysqld_safe sets it to BASEDIR, the MySQL base installation directory.
- DATADIR represents the MySQL data directory. As used to find mysqld-auto.cnf, its default value is the data directory location built in when MySQL was compiled, but can be changed by --datadir specified as an option-file or command-line option processed before mysqld-auto.cnf is processed.