Command Packet

Once the authentication is complete, the client begins sending commands to the server using command packets. The body of a command packet is documented in Table 4-6.

Table 4-6. Format of client command packet

Offset in the body	Length	Description
0	1	Command code.
1	For the noncompressed packet, total packet length from the header – 1. For the compressed packet, the compressed body length – 1.	The argument of the command, if present.

The command codes are contained in `enum server_command`, defined in *include/mysql_ com.h*. The command-handling logic can be found in the `switch` statement of `dispatch_command( )` in *sql/sql_parse.cc*.

Understanding MySQL Internals by Sasha Pachev

COM_SLEEP	0	No argument.	Never sent by a client. Reserved for internal use.
COM_QUIT	1	No argument.	Tells the server to end the session. Issued by the client API call `mysql_close( )`.
COM_INIT_DB	2	A string containing the name of the database.	Tells the server to change the default database for the session to the one specified by the argument. Issued by the client API call `mysql_select_db( )`.
COM_QUERY	3	A string containing the query.	Tells the server to run the query. Issued by the client API call `mysql_query( )`.

COM_FIELD_LIST	4	A string containing the name of the table.	Tells the server to return a list of fields for the specified table. This is an obsolete command still supported on the server for compatibility with old clients. Newer clients use the SHOW FIELDS query.
COM_CREATE_DB	5	A string containing the name of the database	Tells the server to create a database with the specified name. This is an obsolete command still supported on the server for compatibility with old clients. Newer clients use the CREATE DATABASE query.
COM_DROP_DB	6	A string containing the name of the database.	Tells the server to drop the database with the specified name. This is an obsolete command still supported on the server for compatibility with old clients. Newer clients use the DROP DATABASE query.

COM_REFRESH	7	A byte containing the bit mask of reloading operations.	Tells the server to refresh the table cache, rotate the logs, reread the access control tables, clear the host name lookup cache, reset the status variables to 0, clear the replication master logs, or reset the replication slave depending on the options in the bit mask. Issued by the client API call `mysql_refresh( )`.
COM_SHUTDOWN	8	No argument.	Tells the server to shut down. Issued by the client API call `mysql_shutdown( )`.
COM_STATISTICS	9	No argument.	Tells the server to send back a string containing a brief status report. Issued by the client API call `mysql_stat( )`.

COM_PROCESS_INFO	10	No argument.	Tells the server to send back a report on the status of all running threads. This is an obsolete command still supported on the server for compatibility with old clients. Newer clients use the `SHOW PROCESSLIST` query.
COM_CONNECT	11	No argument.	Never sent by a client. Used for internal purposes.
COM_PROCESS_KILL	12	A 4-byte integer with the low byte first containing the MySQL ID of the thread to be terminated.	Tells the server to terminate the thread identified by the argument. Issued by the client API call `mysql_kill( )`. This is an obsolete command still supported on the server for compatibility with old clients. Newer clients use the `KILL` query.

COM_DEBUG	13	No argument.	Tells the server to dump some debugging information into its error log. Issued by the client API call `mysql_dump_debug_info( )`.
COM_PING	14	No argument.	Tells the server to respond with an OK packet. If the server is alive and reachable, it will. Issued by the client API call `mysql_ping( )`.
COM_TIME	15	No argument.	Never sent by a client. Used for internal purposes.
COM_DELAYED_INSERT	16	No argument.	Never sent by a client. Used for internal purposes.

COM_CHANGE_USER	17	A byte sequence in the following format: zero-terminated user name, encrypted password, zero-terminated default database name.	Tells the server the client wants to change the user associated with this session. Issued by the client API call `mysql_change_user( )`.
COM_BINLOG_DUMP	18	A byte sequence in the following format: 4-byte integer for the offset, 2- byte integer for the flags, 4-byte integer for the slave server ID, and a string for the log name. All integers are formatted with the low byte first.	Tells the server to send a continuous feed of the replication master log events starting at the specified offset in the specified log. Used by the replication slave, and in the *mysqlbinlog* command-line utility.
COM_TABLE_DUMP	19	A byte sequence in the following format: 1 byte for database name length, database name, 1 byte for table name length, table name.	Tells the server to send the table definition and data to the client in raw format. Used when a replication slave receives a `LOAD DATA FROM MASTER` query.

COM_CONNECT_OUT	20	No argument.	Never sent by a client. Used for internal purposes.
COM_REGISTER_SLAVE	21	A byte sequence in the following format: a 4-byte integer for the server ID, then a sequence of 1 byte-length prefixed strings in the following order: slave host name, slave user to connect as, slave user password. Then a 2-byte slave user port, 4-byte replication recovery rank, and another 4-byte field that is currently unused. All integers have the low byte first.	Tells the replication master server to register the slave using the information supplied in the argument. This command is a remnant of the started fail-safe replication project. It was introduced in the early version 4.0, but not much has changed since. It is possible that this command might get removed in the future versions.
COM_PREPARE	22	A string containing the statement.	Tells the server to prepare the statement specified by the argument. Issued by the client API call `mysql_stmt_prepare( )`. New in version 4.1.

COM_EXECUTE	23	A byte sequence in the following format: 4-byte statement ID, 1 byte for flags, and 4-byte iteration count. All integers have the low byte first.	Tells the server to execute the statement referenced by the statement ID. Issued by the client API call `mysql_stmt_ execute( )`. New in version 4.1.
COM_LONG_DATA	24	A byte sequence in the following format: 4 byte statement ID, 2 byte parameter number, parameter string. Both integers have the low byte first.	Tells the server the packet contains the data for one bound parameter in a prepared statement. Used to avoid unnecessary copying of a large amount of data when the value of the bound parameter is very long. Issued by the client API call `mysql_stmt_send_long_data( )`. New in version 4.1.
COM_CLOSE_STMT	25	4-byte statement ID with the low byte first.	Tells the server to close the prepared statement specified by the statement ID. Issued by the client API call `mysql_stmt_close( )`. New in version 4.1.

COM_RESET_STMT	26	4-byte statement ID with the low byte first.	Tells the server to discard the current parameter values in the prepared statement specified by the statement ID that may have been set with COM_LONG_DATA. Issued by the client API call mysql_stmt_reset( ). New in version 4.1.
COM_SET_OPTION	27	2-byte code for the option, low byte first.	Tells the server to enable or disable the option specified by the code. At this point, seems to be used only to enable or disable the support of multiple statements in one query string. Issued by the client API call mysql_set_server_option( ). New in version 4.1.
COM_END	28	No argument.	Never sent by a client. Used for internal purposes.