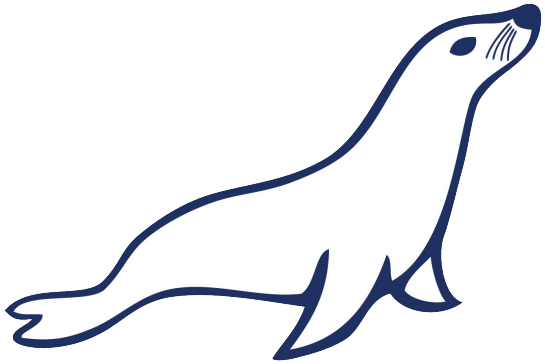

MariaDB®
FOUNDATION

Postmortem: Stack Traces are not resolved in MariaDB

Vicențiu Ciorbaru
Software Engineer @ MariaDB Foundation
vicentiu@mariadb.org

MDEV-14229: Stack trace is not resolved for shared objects

- What goes wrong?
- During a crash:
- In 5.5 line number resolution with addr2line doesn't work at all
- In 10.0+ line number resolution only works for mysqld binary, not for shared objects

MDEV-14229: Stack trace is not resolved for shared objects

```
../sql/mysqld(my_print_stacktrace+0x35)[0x5642209af533]
../sql/mysqld(handle_fatal_signal+0x33f)[0x5642204b69d2]
/lib/x86_64-linux-gnu/libpthread.so.0(+0x11f50)[0x7fba0369af50]
../mysql-test/var/plugins/adt_null.so(+0x97a)[0x7fb9fe44397a]
../sql/mysqld(_Z23initialize_audit_pluginP13st_plugin_int+0x8c)[0x564220404c2c]
../sql/mysqld(+0x3efd4b)[0x564220319d4b]
../sql/mysqld(+0x3f18cc)[0x56422031b8cc]
../sql/mysqld(_Z20mysql_install_pluginP3THDPK19st_mysql_lex_stringS3_+0x2d6)[0x56422031bdc7]
../sql/mysqld(_Z21mysql_execute_commandP3THD+0x6bcf)[0x56422030d73e]
../sql/mysqld(_Z11mysql_parseP3THDPcjP12Parser_state+0x210)[0x5642203110dd]
../sql/mysqld(_Z16dispatch_command19enum_server_commandP3THDPcj+0xc26)[0x56422030498f]
../sql/mysqld(_Z10do_commandP3THD+0x2cb)[0x564220303b91]
../sql/mysqld(_Z24do_handle_one_connectionP3THD+0x1dc)[0x564220407c9c]
../sql/mysqld(handle_one_connection+0x33)[0x564220407a15]
../sql/mysqld(+0x8607f2)[0x56422078a7f2]
/lib/x86_64-linux-gnu/libpthread.so.0(+0x75aa)[0x7fba036905aa]
/lib/x86_64-linux-gnu/libc.so.6(clone+0x3f)[0x7fba0265dcbf]
```

MDEV-14229: Stack trace is not resolved for shared objects

```
mysys/stacktrace.c:246(my_print_stacktrace)[0x561c94c845e3]
sql/signal_handler.cc:155(handle_fatal_signal)[0x561c9478ba82]
/lib/x86_64-linux-gnu/libpthread.so.0(+0x11f50)[0x7ff049b56f50]
audit_null/audit_null.c:57(audit_null_plugin_init)[0x7ff0448ff97a]
sql/sql_audit.cc:379(initialize_audit_plugin(st_plugin_int*))[0x561c946d9cdc]
sql/sql_plugin.cc:1376(plugin_initialize(st_mem_root*, st_plugin_int*, int*, char**,
bool))[0x561c945eedfb]
sql/sql_plugin.cc:2035(finalize_install(THD*, TABLE*, st_mysql_lex_string const*, int*,
char**))[0x561c945f097c]
sql/sql_plugin.cc:2136(mysql_install_plugin(THD*, st_mysql_lex_string const*,
st_mysql_lex_string const*))[0x561c945f0e77]
sql/sql_parse.cc:4409(mysql_execute_command(THD*))[0x561c945e27ee]
sql/sql_parse.cc:5923(mysql_parse(THD*, char*, unsigned int,
Parser_state*))[0x561c945e618d]
sql/sql_parse.cc:1068(dispatch_command(enum_server_command, THD*, char*, unsigned
int))[0x561c945d9a3f]
sql/sql_parse.cc:793(do_command(THD*))[0x561c945d8c41]
sql/sql_connect.cc:1268(do_handle_one_connection(THD*))[0x561c946dcd4c]
sql/sql_connect.cc:1185(handle_one_connection)[0x561c946dcac5]
perfschema/pfs.cc:1017(pfs_spawn_thread)[0x561c94a5f8a2]
nptl/pthread_create.c:463(start_thread)[0x7ff049b4c5aa]
x86_64/clone.S:97(clone)[0x7ff048b19cbf]
```

Steps to solve a bug

1. Reproduce
2. Analyze under debugger
3. Find a solution!

Background

- MariaDB has many ways to attempt to resolve stacktraces.
- Most common one is with addr2line
- Current implementation will fork addr2line and pass it the list of addresses for all frame pointers.
- We will wait for addr2line to respond with a source file and line number based off of the address.
- Why does it not work? -> Debugger!

What goes wrong?

- We get the list of frame addresses via backtrace() function correctly.
- We pass a frame address to addr2line correctly
- But addr2line returns ??, why?
  - The frame address is not the actual address in the binary
    - Binaries get loaded at an offset in virtual memory
  - Addr2line is only started for mysqld binary. We need to start it for each .so file.

How do we fix it?

- 2 problems
  - Identify from which binary a function address comes from and what is the offset at which the binary is loaded

    dladdr() can help

  - Start addr2line for each identified binary

Thank You!

Contact me at:
vicentiu@mariadb.org

Blog:
mariadb.org/blog