gdb tips and tricks for MySQL DBAs

or

How gdb can help you to solve MySQL problems

Valerii Kravchuk, Principal Support Engineer
valerii.kravchuk@percona.com
Who am I?

Valerii (aka Valeriy) Kravchuk:

- MySQL Support Engineer in MySQL AB, Sun and Oracle, 2005 - 2012
  - Bugs Verification Team all this time
  - Support issues related to bugs, crashes, InnoDB, performance...
  - Trainings (mostly informal) for new team members
  - All kinds of decision making committees...
- Principal Support Engineer in Percona since September, 2012
  - Doing more or less the same as before, but better (I hope)...
  - Plus I try to speak and write about MySQL in public now
- [http://mysqlentomologist.blogspot.com](http://mysqlentomologist.blogspot.com) - my blog about MySQL (mostly bugs)
- [https://www.facebook.com/valerii.kravchuk](https://www.facebook.com/valerii.kravchuk) - my Facebook page, a lot about MySQL (mostly bugs…)
- [http://bugs.mysql.com](http://bugs.mysql.com) - my personal playground. 6 bugs reported in 2015
What is this session about?

- Some historical remarks and URLs to known use cases/blog posts about gdb and MySQL troubleshooting
- Multi-threaded executables and gdb (threads, frames)
- Basic gdb commands and “tricks”
- Few words on pt-pmp use
- Important MySQL data structures to explore (mostly THD *)
- A couple of real life use cases, working with core dump and alive mysqld
- Discussion
Domas is famous for these tricks...

- [http://dom.as/2010/01/02/read-ahead/](http://dom.as/2010/01/02/read-ahead/)
- [http://dom.as/2009/03/14/stupid-innodb-tricks/](http://dom.as/2009/03/14/stupid-innodb-tricks/) - not literally needed since 5.1.38 and InnoDB plugin, innodb_spin_wait_delay :)
- [http://poormansprofiler.org/](http://poormansprofiler.org/)

More examples of gdb use for MySQL DBAs

- Remember the names: Domas Mituzas, Shane Bester, Roel Van De Paar, Mark Callaghan, Aurimas, Zhai Weixiang, ...
What you can do with gdb

- Check stack traces (and variables), per thread:
  thread apply all bt [full]
- Print variables, up to complex one:
  thread 1
  print do_command::thd->query_string.string.str
- Set new values for variables (global and per thread, even those formally read-only in MySQL while it’s running):
  set max_connections=5000
  set opt_log_slave_updates=1
- Call functions (that may do complex changes):
  call rpl_filter->add_do_db(strdup("hehehe"))
- Set breakpoints and watchpoints
- Work interactively or use **gdb** as a command line utility (**-batch**)
- Use macros, Python scripting, more…
- All these may not work, fail, hang, crash, produce obscure errors…
- You have to read and understand the source code
pt-pmp (Poor Man’s Profiler)

- [link](http://www.percona.com/doc/percona-toolkit/2.2/pt-pmp.html)


- It is based on original idea by Domas ([http://poormansprofiler.org/](http://poormansprofiler.org/)) with some more bash/awk on top applied.
- When mysqld hangs or is slow, you can get some insight quickly: [http://bugs.mysql.com/bug.php?id=75028](http://bugs.mysql.com/bug.php?id=75028) (HandlerSocket “hangs” on shutdown)
- pt-pmp surely **slows server down** :) Hint (workaround is in the bug): [https://bugs.launchpad.net/percona-toolkit/+bug/1320168](https://bugs.launchpad.net/percona-toolkit/+bug/1320168)
Multi-threaded mysqld and gdb

- process/thread/frame concepts:

  (gdb) thread 2
  [Switching to thread 2 (Thread 0x7fe771550700 (LWP 2544))]
  #0 0x00000000000605774 in Item_func_numhybrid::val_int (this=<value optimized out>)
      at /home/openxs/bzr2/percona-5.6/sql/item_func.cc:1013
  1013   }
  (gdb) bt
  ...
  #12 0x000000000006f8a45 in dispatch_command (command=COM_QUERY,
      thd=0x7fe760f94000, packet=0x7fe77154fac0 "", packet_length=0)
      at /home/openxs/bzr2/percona-5.6/sql/sql_parse.cc:1434
  ...
  (gdb) frame 12
  #12 0x000000000006f8a45 in dispatch_command (command=COM_QUERY,
      thd=0x7fe760f94000, packet=0x7fe77154fac0 "", packet_length=0)
      at /home/openxs/bzr2/percona-5.6/sql/sql_parse.cc:1434
  warning: Source file is more recent than executable.
  1434   mysql_parse(thd, thd->query(), thd->query_length(), &parser_state);
  (gdb) p thd->query_string.string.str
  $2 = 0x7fe75301d010 "select benchmark(5", '0' <repeats 13 times>, ",", 2*2)"

- https://sourceware.org/gdb/onlinedocs/gdb/Variables.html
THD structure

grep -rn THD sql/sql_class.h

class THD :public MDL_context_owner,

            public Statement,

            public Open_tables_state

HASH user_vars; // hash for user variables

struct system_variables variables; // Changeable local variables

struct system_status_var status_var; // Per thread statistic vars

struct system_status_var *initial_status_var; /* used by show status */

Security_context main_security_ctx;

...

CSET_STRING query_string; // inherited from Statement...
THD structure (continued)

(gdb) p thd->main_security_ctx->user
$7 = 0x7fe753019058 "root"

(gdb) p thd->main_security_ctx->host
$8 = {Ptr = 0xc16759 "localhost", str_length = 9, Allocated_length = 0,
     allocated = false, str_charset = 0x1393de0}
Real life case: checking core dump

```
gdb -ex 'set pagination 0'\  
...\  
  -ex 'thread apply all bt full'\  
/path/to/mysqld /var/tmp/core.<pid> | tee core.<pid>.bt
```

- Make sure you know how to get core when mysqld crashes:


- Let’s check one example, we need crashing bug for this:

  https://bugs.launchpad.net/percona-server/+bug/1384658
Real life case: attaching to alive mysqld

This is how it goes:

```bash
[root@centos openxs]# mysql -uroot -e "show variables like 'innodb_autoinc_lock_mode'"
+--------------------------+-------+
| Variable_name            | Value |
+--------------------------+-------+
| innodb_autoinc_lock_mode | 0     |
+--------------------------+-------+
[root@centos openxs]# mysql -uroot -e "set global innodb_autoinc_lock_mode=1"
ERROR 1238 (HY000) at line 1: Variable 'innodb_autoinc_lock_mode' is a read only variable
[root@centos openxs]# gdb -ex "set innobase_autoinc_lock_mode=1" -batch -p `pidof mysqld`
...  
[Thread debugging using libthread_db enabled]
0x00007ff31d6830d3 in poll () from /lib64/libc.so.6
... check the variable value again now

[root@centos openxs]# ps aux | grep mysqld
[root@centos openxs]# kill -SIGCONT `pidof mysqld`
```
Thank you!

Questions and Answers

Please, report bugs at:
http://bugs.mysql.com
Use “Affects Me” button!