How to get your bug fixed fast

Oleksandr (Sanja) Byelkin
Principal Software Engineer
MariaDB Corporation

Disclaimer: It is a developer’s point of view
How?

Invest:

- Money (support)
  - Help with preparing test case
  - Prioritising support bugs
- Time (prepare excellent test case, it is our topic)
  - Developer will be immediately ready to start fixing
  - Higher chance to be chosen earlier
- Both!
Main thing developer want from you
Main thing developer want from you

HOW TO REPRODUCE THE BUG?
(a test case)
What is important to repeat it (most frequently)

- Query (script, exact steps to reproduce etc.)
- Database (dump of all related structures)
- Expected vs received results or stack trace
- Server version
- Server configuration
  - Operating system
  - Environment variables
  - Day time and phase of the moon
Where to get query

- You made it (easiest but rare case)
- Logs (most realistic way)
  - General query log has impact on performance (switch it temporary or on non-production database)
  - It is not the last query in the log which caused the problem, but one of the last (last on problematic connection)
- Crash report in your error log
Database issue logging

- Confidential information (everything attached to the issue is public, use foundation ftp if it is confidential)
- Size

We are happy to get test case of any size and promise to not sell data, but better be safe than sorry…
How to solve the database problems

- Use schema only (it is not the best variant, in this case it is better also describe data distribution)
- Fill the schema with random or generated data (maybe you have test/demo data for application development) and can reproduce the problem.
- For more security after repeating you can rename tables/fields/procedures/indices in the schema and the query
  - Tables t1, t2, t3...
  - Fields f1, f2, f3...
  - Procedures p1, p2, p3...
  - ...
Frequent mistakes I saw

- Trying to prove that bug exists instead of showing how to repeat it
- Detailed description of not so important details like operating system, standard configuration, but close to nothing about how to repeat bug
- Only stack trace (sometimes it is enough but most frequently no)
  - Also core dump with matching mysqld binary and ldd files will be good addition to the stack trace (for ldd see https://github.com/mariadb-corporation/mariadb-qa/blob/master/ldd_files.sh)
Reporting a bug

https://jira.mariadb.org

System Dashboard

Introduction

MariaDB's JIRA Instance
## Summary, Priority and Affected Version/s

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summary*</td>
<td>Sum() produce incorrect result</td>
</tr>
<tr>
<td>Priority</td>
<td>Major</td>
</tr>
<tr>
<td>Component/s</td>
<td></td>
</tr>
<tr>
<td>Affects Version/s</td>
<td>10.1.46</td>
</tr>
<tr>
<td>Environment</td>
<td></td>
</tr>
</tbody>
</table>
Environment, Description and Attachment

**Environment**

Ubuntu 20.04

For example operating system, software platform and/or hardware specifications (include as appropriate)

**Description**

In uploaded example database

```sql
SELECT SUM(*) FROM T1 WHERE F2=3  
```

we get 5 instead of 4.

**Attachment**

Drop files to attach, or browse.
Create

Affects Version/s: 10.1.46

Environment: Ubuntu 20.04

Description:

In uploaded example database
{code}
SELECT SUM(*) FROM T1 WHERE F2=3
{code}
we get 5 instead of 4.
Fields

Important

- Project
- Issue Type
- Summary
- Description
- Attachments (Always public, use mariadb foundation FTP server for something private or secret

Can be left as is but...

- Priority (default is OK, you can use it to point how it is important for you, but later it will be used for our priority)
- Component/s (difficult to find without deep analysis, just leave as is if unsure)
- Environment (very rare it can be helpful, you can put OS here, no need spend long time on it)
- … other fields (usually used for the issue handling)
Communication

You will get emails with changes in your bug reports.

- Comments - they used to communicate between developers or support, internal notes in bug processing, but also to ask reporter questions (it is very important).
- Label. It is important for you to notice if the 'need_feedback' label is added to the bug report it means that someone need your feedback.
If you have time and inspiration

- Before any improvement save your current repeatable test case
- Reduce database
  - Remove unused tables
  - Remove unused fields and indices
- Simplify the query (leave only important parts)
  - Tables
  - Clauses
  - Fields
  - Functions
  - ...

MariaDB