



Toolforge webservices are in the final stages of [migrating to the toolforge.org domain](#).  
Please help us clean up older documentation referring to tools.wmflabs.org!

# Incident documentation/20170728-s5 (WikiData and dewiki) read-only

[< Incident documentation](#)

Wikidata and the German Wikipedia were in read-only mode from 5:48 to 7:04 on 2017-07-28. No edits, user creations were possible between these two timestamps.

## Contents [\[hide\]](#)

- [1 Summary](#)
- [2 Timeline](#)
- [3 Conclusions](#)
- [4 Actionables](#)
  - [4.1 Short term](#)
  - [4.2 Long term](#)

## Summary

The s5 master MySQL process (containing wikidatawiki and dewiki) was killed at 5:48 due to an Out Of Memory error- which lead to the server automatically reboot in read only, recover its data and stay like that until it was noticed, checked and set back as read-write.

## Timeline

- 05:48:07: db1063 kernel runs out of memory and choses the process using the most memory to be killed (obviously), MySQL:

```
Jul 28 05:48:07 db1063 kernel: [7566480.043193] Free swap = 0kB
Jul 28 05:48:07 db1063 kernel: [7566480.043275] Out of memory: Kill process
4744 (mysqld) score 757 or sacrifice child
Mediawiki connection errors to the master start at this time.
Cannot access the database: Lost connection to MySQL server at 'waiting for
initial communication packet', system error: 110 (10.64.32.228)
```

- 05:48:14: db1063 mysqld\_safe watchdog notices the MySQL process is gone, and restarts the server:

```
Killed
170728 05:48:14 mysqld_safe Number of processes running now: 0
170728 05:48:14 mysqld_safe mysqld restarted
170728 5:48:14 [Note] /opt/wmf-mariadb10/bin/mysqld (mysqld 10.0.29-MariaDB)
starting as process 23880 ...
```

- 05:49:31: The server performs recovery cleanly, and starts accepting connections again, but by default a server on restart is in (intentionally) in read-only for 2 reasons- if the server starts back in a bad state, we will not want to server to be pooled again as master, and perform a failover instead -so a human gives the ok to continue; also, on boot, the server will be with an empty buffer pool, which in some cases may be worse than not accepting connections at all. Last mediawiki exception connection happens at 2017-07-28T05:49:30.
- 05:51:10: Mediawiki detects when a database master is in read only and sets itself in read-only, too, to give better feedback to users and avoid unnecessary errors. 05:51:10 is (with some few other errors, like the inability of locking) when mediawiki start to send read-only exceptions. Errors are low because Mediawiki discourages and/or makes impossible from doing edits to users.
- 06:12 First report is sent to IRC "<SMalyshev> wikidata changes stopped about 20 mins ago - anybody knows the reason?", but no one that can take action sees it :-(. Because not many errors/exceptions happen- users get nice feedback because of the read only mode, it is most likely that most users think it is a normal

[Main page](#)  
[Recent changes](#)  
[Server admin log \(Prod\)](#)  
[Server admin log \(RelEng\)](#)  
[Deployments](#)  
[SRE/Operations Help](#)  
[Incident status](#)

[Cloud VPS & Toolforge](#)  
[Cloud VPS documentation](#)  
[Toolforge documentation](#)  
[Request Cloud VPS project](#)  
[Server admin log \(Cloud VPS\)](#)

[Tools](#)  
[What links here](#)  
[Related changes](#)  
[Special pages](#)  
[Permanent link](#)  
[Page information](#)  
[Cite this page](#)

[Print/export](#)  
[Create a book](#)  
[Download as PDF](#)  
[Printable version](#)

maintenance.




- 06:16 Ticket is created on Phabricator <https://phabricator.wikimedia.org/T171928>. Sadly, those hours are the ones with less available operators or developers, so doesn't get quick attention.
- 06:51 First report seen on IRC (#wikimedia-operations)
- 06:58 Jaime notices and starts investigating
- 07:01 Database seems up, replication running, unsure why issues are happening -if they are application or database issues. Database is seen as read-only finally.
- 07:04 After checking database configuration, error log (specially the recovery log - `/srv/sqldata/db1052.err`), server is put back into read-write ( `SET GLOBAL read_only=0` ). **Most wiki issues should stop here.** This is later confirmed with some test edits.
- Following investigation reveals the reason for the crash is an OOM killer, as discovered by moritz. The swap monitoring clearly shows that: <https://grafana.wikimedia.org/dashboard/file/server-board.json?refresh=1m&panelId=16&fullscreen&orgId=1&var-server=db1063&var-network=bond0&from=1498634959402&to=1501226959402>. After further investigation, a memory leak is detected that started back on 5th of July 2017 at 14:00 <https://grafana.wikimedia.org/dashboard/file/server-board.json?refresh=1m&orgId=1&var-server=db1063&var-network=bond0&from=1499174616320&to=1499407392494&panelId=14&fullscreen>. The reason for the leak turned it to be a pt-kill process running since that exact date that was set in place to avoid large long-running admin-user queries from wikidata.

## Conclusions

- pt-kill leaks memory (albeit slowly). All instances of that application should be killed and restarted on a cron or similar to avoid those issues (or substituted with database events, as those seem to work nicely)
- Database master s5 crashed and restarted in read-only mode. This is a known limitation due to masters being SPOF- however, a failover would not have mitigated the issue here- it most likely not have been detected, as MySQL restarts automatically and everything, from replication to the MySQL status appear OK.
- The main issue here was not to detect the read only mode on mediawiki or the database quickly- taking over an hour to react- once it was detected, it took less than 6 minutes to fix. This points to several mitigations options- puppetize masters in read-write (automatic fix, but dangerous should something worse than a kill happens), or better monitoring of the uptime or read\_only value with paging.
- Regarding memory consumption, a swap/memory monitoring would also help detecting the issue well in advance, as it took over 2 months for the memory leak to manifest itself.
- Is there a gap in our human coverage? Could something have been done in terms of coordination at those times?
- As a positive note, because independent resources, a single database going down didn't impact readers, nor editors from the other ~800 projects, including commons, meta, enwiki (even if they were consumers of Wikidata). s5 is planned to be split soon due to the load requirements of Wikidata, hopefully mitigating even more issues between projects.

## Actionables

### Short term

- Monitor read\_only variable and/or uptime on masters, make it page [Task T172489](#)  **In progress**
- Substitute pt-kill for events or restart its execution regularly to avoid memory leaks  **Done**, it is an always ongoing task, but fixed for s5.
- Monitor memory/swap usage on all databases, warn if too much memory pressure in advance [Task T172490](#)  **In progress**

### Long term

- Change pt-heartbeat model to not use super-user, avoid SPOF and switch automatically to the real master [Task T172497](#)
- Change oom\_adj for dedicated mysql server processes [Task T172494](#)
- Change mariadb monitoring tool and model [Task T172492](#)

Category: [Incident documentation](#)

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. See [Terms of Use](#) for details.

[Privacy policy](#) [About](#)

[Disclaimers](#) [Code of Conduct](#) [Developers](#) [Statistics](#) [Cookie statement](#) [Mobile view](#)

[Wikitech](#)

