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

Incident documentation/20200528-commons read-only

< Incident documentation

document status: in-review

Summary

s4 primary database master (db1138) had a hardware memory issue, mysqld process crashed and came back as read-only for 8 minutes.

Impact: commonswiki didn't accept writes for 8 minutes. Reads remained unaffected

Contents [hide]

Timeline

All times in UTC.

- 01:33 First signs of issues on the DIMM are logged on the hosts's IDRAC
- 20:21 mysql process crashes **OUTAGE BEGINS**
- 20:24 First page arrives: PROBLEM - MariaDB read only s4 #page on db1138 is CRITICAL: CRIT: read_only: True, expected False:
- 20:24 A bunch of SREs and a DBA start investigating and the problem is quickly found as a memory DIMM failure and mysql process crashed
- 20:28 <@marostegui> !log Decrease innodb poolsize on s4 master and restart mysql
- 20:29 MySQL comes back and read_only is manually set to OFF
- 20:29 **OUTAGE ENDS**

Conclusions

This was a hard to avoid crash - hardware crash on a memory DIMM.

Masters start as read-only by default (to avoid letting more writes go through after a crash, until we are fully sure data and host are ok and still able to take the master role).

We did see traces of issues on the idrac's error log, if we could alert on those, maybe we could have performed a master failover before this host crashes. If this crash happens on a slave, the impact wouldn't have been as big, as slaves are read-only by default and MW would have depooled the host automatically.

What went well?

- The read_only alert that pages when a master comes back as read-only worked well (this was an actionable from a similar previous issue)
- Lots of people made themselves available to help troubleshooting, from both EU (late UTC evening already) and US (from different TZs)

What went poorly?

- We had signs of this memory failing on the idrac's hw log ardound 19 hours before the actual crash, we could have alerted on those beforehand, scheduled an emergency failover and prevent this incident.

Where did we get lucky?

- The host and the data didn't come back corrupted, otherwise we'd have needed to do a master failover at that same moment

How many people were involved in the remediation?

- 1 DBA even though lots of SRE made themselves available in case they were needed.

Links to relevant documentation

- There is not specific documentation on what to do when a master crashes and comes back as read only. Action point: https://phabricator.wikimedia.org/T253832

Actionables

- [DONE] Documentation on how to proceed if a master pages for read-only = ON: https://phabricator.wikimedia.org/T253832
- [DONE] Failover db1138 to its candidate master (scheduled for Friday 29th at 05:00 AM UTC): https://phabricator.wikimedia.org/T253808
- [DONE] Replace failed DIMM on that host: https://phabricator.wikimedia.org/T253808
- Alert on ECC warnings in SEL https://phabricator.wikimedia.org/T253810
- Create a script to move replicas between hosts when the master isn't available https://phabricator.wikimedia.org/T196366

Categories: Incident documentation | Incident documentation drafts