Page **Discussion**

Read **View source** **View history**

Search Wikitech

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/20180229-Train-1.31.0-wmf.27

< Incident documentation

Contents [hide]

Summary

Train deployment of 1.31.0-wmf.27 rolled back due to a increase in replication wait errors.

Timeline

- 19:24 Synchronized php: group1 wikis to 1.31.0-wmf.26 (duration: 01m 17s)
- 19:22 rebuilt and synchronized wikiversions files: group1 wikis to 1.31.0-wmf.26
- 19:20 Rolling back to wmf.26 due to increase in fatals: "Replication wait failed: lost connection to MySQL server during query"
- 19:19 rolling back to wmf.26
- 19:18 icinga-wm: PROBLEM - MediaWiki exceptions and fatals per minute on graphite1001 is CRITICAL: CRITICAL: 80.00% of data above the critical threshold [50.0]
- 19:17 twentyafterfour: | I'm seeing quite a few "[{exception_id}] {exception_url} Wikimedia\Rdbms\DBExpectedError: Replication wait failed: Lost connection to MySQL server during query
- 19:06 twentyafterfour@tin: Synchronized php: group1 wikis to 1.31.0-wmf.27 (duration: 01m 17s)
- 19:05 twentyafterfour@tin: rebuilt and synchronized wikiversions files: group1 wikis to 1.31.0-wmf.27

Error graph from the same time period: https://grafana.wikimedia.org/dashboard/db/mediawiki-graphite-alerts?orgId=1&panelId=2&fullscreen&from=1522263260081&to=1522265839537

After rollback Mukunda filed Task T190960 to document the incident. The culprit was determined to be this commit which was made by Aaron Schulz. That commit was intended to address the issues described in Task T180918.

Conclusions

What weakness did we learn about and how can we address them?

It is exceedingly difficult to thoroughly test some changes outside of production. Testing replication lag detection properly would require a simulation of production databases plus realistic traffic to stress them. Our current deployment process prevented this from having an impact on production databases or site reliability, however, we spent a lot of time deploying and then reverting changes and we blocked testing of other changes in the pipeline. This points to weaknesses in the weekly branching strategy that we currently employ as well as weaknesses in our testing environments. A change such as this one should really have it's own staged deployment process rather than "riding the train" concurrent with a bunch of unrelated changes.

Actionables

NOTE: Please add the #wikimedia-incident Phabricator project to these follow-up tasks and move them to the "follow-up/actionable" column.

- phab:T193258

Category: Incident documentation

This page was last edited on 19 October 2018, at 11:54.