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/20140820-SwiftMemcache

< Incident documentation

Summary

Starting 2014-08-19 multiple users reported intermittent failures when doing operations on files (esp. deletes in commons) resulting in bugzilla:69760 to be opened. Investigation into swift later revealed several timeouts while talking to memcache.

Timeline

- 2014-08-19T20:54:57Z bugzilla:69760 filed by users reporting errors
- 2014-08-20T13:50:15Z bug escalated and investigated. An XFS issue on ms-be1003 was found as probably root cause and the machine rebooted, seemingly fixing the problem.
- 2014-08-20T17:45:59Z more issues of the same type reported on the bug, however the failures were intermittent with users usually succeeding upon retry.
- 2014-08-21T20:54:55Z after further investigation into the swift logs several timeouts from swift proxy talking to memcache have been observed
- 2014-08-21T21:33:55Z https://gerrit.wikimedia.org/r/#/c/155629/ is deployed to increase maximum connections to memcache, thus fixing the frequent timeouts

Conclusions

- The interactions between mw and swift have been historically troublesome and opaque to users, this has been no different.
- The root cause was thought to be fixed by restarting a machine in trouble, however that was a red herring.
- The real resolution was slowed down in part by the amount of logging swift does as part of its operation, masking potential problems. Another contributing factor was lack of deep insight into how mw and swift interact and what mw metrics are related to that.
- Upon inspection the swift code involved in talking to memcache doesn't seem to provide any instrumentation and thus no metrics.
- The default connection limit of 2 connections per worker was obviously not adequate with 4 memcache machines, however this doesn't seem to have caused such widespread problems in the past (compounded by the lack of error visibility too)

Actionables

- Status: **Declined** - separate swift request logging from error logging RT #8181
- Status: **Declined** - instrument swift memcache code to report operational metrics RT #8182
- Status: **Declined** - improve visibility on the interactions between mw and swift and what metrics are affected

Category: Incident documentation