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/20170419-ContentTranslation

< Incident documentation

Summary

CX caused an outage on one of the database servers affecting CX, Echo and Flow. The x1 database was overloaded with long-running FOR UPDATE queries until it had to be forcefully restarted.

Likely root cause was a bug in the frontend code that in certain articles caused the save draft request size to be extra large due to inclusion of unrelated content combined with an unoptimal autosave-retry logic exacerbating the problem to an outage.

Timeline

- [14:00] Data center switch starts
- [14:32] First lonely error in Logstash about CX (observed later)
- [14:40] Lots of errors in Logstash about CX (observed later)
- [15:13] Language team is pinged to assist with an issue with queries generated by CX
- [15:30] CX queries are being killed, but connections do not get disconnected
- [15:35] Disabling of CX suggested
- [15:36] Outage on x1 started as limit of open connections was exceeded
- [15:42] CX is disabled
- [15:46] More load to slave, read-only triggered
- [15:57] Database is restarted, recovery starts
- [16:00] T163344 filed
- [16:18] Outage over
- [17.33] Annoucement to wikitech-l that datacenter switch is complete, mentioning that CX is disabled

Investigation

Thursday

- Language team starts investigating frontend and backend
- Finds out that the front-end can do more requests than expected
- Finds out that the ping-limiter does not help to prevent query build-up in cases like this

Friday

- Patch to improve auto-saving to be much more conservative was merged
- A list of long queries running during the outage is provided
- There is a mixed data about number of save requests that happened during the outage: EventLogging and Grafana don't show spikes, even though there were many queries during the outage

Monday

- A disk IO overload is suspected as the cause, quickly proven false

Tuesday

- Focus is now on understanding what caused the locking issue
- CX was re-enabled and database was monitored more closely

Wednesday

- Database performance was monitored through out the day and found no issues
- A bug in the frontend code was found by analyzing a particular query was found repeating 1000+ times during the incident T163105

Thursday

- T163105 was further investigated to see that it can cause a database failure combined with the autosave-retry logic.

Conclusions

- The CX frontend is very simple, not understanding if an outage is happening, so it might have made things worse with automatic retries.
- Database locking is complex. The current saving code was written by another person, not fully understood by the language team.
- Language team was not monitoring the status during the switchover. It should have.
- The outage affected not only CX, but also Flow and Echo. The good thing is that only those services were mainly affected. The bad thing is that other services than CX were affected. A ways to have unrelated service outages not affect each other should be investigated.
- Ways to fail faster (detect and act on blocked queries) before it escalates to an outage should be investigated.

Actionables

- Status: **Done** Improve the autosave-retry logic. Part of T163344
- Status: **Done** Fix the bug that caused malformed safe requests. T163105
- Status: **Dropped** Enforce size limits for saved blobs. T164050
- Status: **Done** Be stricter about slow queries. T160984
- Status: **Done** Audit the queries for possible lock issues. Part of T163344

Also see

- https://wikitech.wikimedia.org/wiki/Incident_documentation/20160713-ContentTranslation

Category: Incident documentation