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/20150527-Cookie

< Incident documentation

2015-05-27 Cookie incident

Contents [hide]

Summary

MediaWiki 1.26wmf7 was deployed to group2 wikis (bulk of our traffic) as part of our normal deploy train. It contained https://gerrit.wikimedia.org/r/#/c/176948/ , which introduced new cookie names for not-logged-in users matching the session cookie regexes in our Varnish layer, which killed cache performance in general and caused an elevated level of 503 errors for the duration of the incident and increased load elsewhere in the stack. The incident lasted approximately 5 hours. The sites were generally up and usable during this time, but users noted decreased performance and increased odds of random 503 errors. There was also a ~5 minute full outage in the midst of the incident due to a process failure while trying to revert the 1.26wmf7 deploy.

Timeline (UTC)

- [20:41:55] <logmsgbot> !log twentyafterfour rebuilt wikiversions.cdb and synchronized wikiversions files: wikipedias to 1.26wmf7
- . .. ~15 mins for resourceloader cache to expire and bring the problem to a head, then graphite first notes the 503 anomalies ...
- [21:04:08] <icinga-wm> PROBLEM - HTTP 5xx req/min on graphite1001 is CRITICAL 33.33% of data above the critical threshold [500.0]
- [21:09:39] <icinga-wm> PROBLEM - HTTP error ratio anomaly detection on graphite1001 is CRITICAL Anomaly detected: 11 data above and 1 below the confidence bounds
- [21:11:50] <bblack> what's going on?
- ... some investigation of earlier deployment issues, which were not related/causative/significant ...
- ... first individual report of ongoing issues:
- [21:26:29] <Danny_B> Error: 503, Service Unavailable at Wed, 27 May 2015 21:26:03 GMT
- ... graphite continues complaining:
- [21:29:49] <icinga-wm> PROBLEM - HTTP 5xx req/min on graphite1001 is CRITICAL 21.43% of data above the critical threshold [500.0]
- ... several hhvm TC issues reported of the form:
- [21:30:19] <icinga-wm> PROBLEM - Translation cache space on mw1250 is CRITICAL: HHVM_TC_SPACE CRITICAL code.main: 91%
- ... 40 mins time burned with bblack and ori investigating, mostly debating whether the TC cache issues are significant
- ... first start noticing real evidence that the problem is in fact much more serious than that:
- [22:09:00] <bblack> anyways, I think 5xx is still ongoing regardless of TC spike or TC-related restart
- [22:09:12] <bblack> I can see backend health issues at the varnish level too, but I'm not sure if they're indirect...

From here on, hours of complex debugging/analysis ensue, mostly logged in #wikimedia-operations, involving at various points in time: bblack, ori, MaxSem, legoktm, twentyafterfour, Reedy, marktraceur, and probably a few others I'm forgetting. Most of this is focused on a red herring bblack keeps pointing out about abormal query rates and failures for **/w/api.php?action=query&format=json&meta=filerepoinfo&smaxage=86400&maxage=86400**, which turned out to just be a victim/symptom rather than the real cause.

Eventually we attempted to revert the most recent deploy train updates, as the timing of their deployment aligned with the anomalies and no other clear cause had yet been deduced:

- [22:52:27] <logmsgbot> !log twentyafterfour rebuilt wikiversions.cdb and synchronized wikiversions files: roll back everything but testwiki to 1.26wmf6

However, there was a process error with the reversion itself leading to l10n cache errors taking us into a complete outage meltdown for about a 5 minute period until ori reverted the revert:

- [22:57:33] <logmsgbot> !log ori rebuilt wikiversions.cdb and synchronized wikiversions files: (no message)

After investigating this process failure, we again reverted the wmf7 deploy:

- [23:05:24] <grrrit-wm> (Merged) jenkins-bot: lets try 1.26wmf6 again, this time with l10ncache [mediawiki-config] - https://gerrit.wikimedia.org/r/214266 (owner: 20after4)

More debugging ensued from here, as we were still experiencing approximately the same problems in spite of the revert. Eventually during the investigation, legoktm pointed out a cookie-affecting commit contained in the 1.26wmf7 changes:

- [00:38:44] <legoktm> cookie related there was https://gerrit.wikimedia.org/r/#/c/176948/

After some discussion/review it became apparent that commit was a very likely candidate. While this commit had already been reverted as part of the wmf7 revert earlier, because the mechanism of action was to set a very performance-destructive cookie, the cookies persisted in our users' browsers and kept us from observing any real recovery from the revert. Eventually we merged up a varnish-level fixup to work around the broken cookies still being sent by clients:

- [01:08:05] < grrrit-wm> (CR) BBlack: [C: 2 V: 2] Negative lookbehind for mwuser-session cookies [puppet] - https://gerrit.wikimedia.org/r/214281 (owner: BBlack)

After a few minutes to get this deployed (salt failures slowed deploying the fix to the varnishes), everything returned to normal, with confirmation of the fix in stats/health by ~01:20

Conclusions

- The primary cause was the merging and deployment of a code change which created a new cookie name which inadvertently wreaked havoc on our Varnish caching layers.
- Confusing processes for reverting MW deployments exacerbated the impact of the outage.
- A similar incident occured back in Apr 2014, after which we supposedly implemented a rule that ops should code-review any cookie changes prior to deployment to prevent exactly this sort of scenario. No such review seems to have occurred with this change. Such a review almost certainly would have caught the issue, as it's fairly obvious that this commit would will cause this problem from an ops perspective. In defense of the commit, I do not believe this rule was ever codified in any written coding/deployment practices anywhere, and so it's quite likely that this lesson was becoming lost to our organizational memory and the committers were unaware of it.

Actionables

- Status: 🟥 **Unresolved** – Formally document ops review of cookie changes prior to WMF deployment of new MediaWiki versions and take measures to ensure this happens in practice
  - Reconsider the approach being taken with cookies and caching – phabricator:T100920
- Status: 🟥 **Unresolved** – Make sync-wikiverisons check that a valid localisation cache exists when syncing new versions – phabricator:T100573
- Status: 🟥 **Unresolved** – Ability to scap only one version – phabricator:T100575

Category: Incident documentation

This page was last edited on 31 May 2015, at 06:35.