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

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/20200211-caching-proxies

< Incident documentation

document status: final

Contents [hide]

Summary

During maintenance on our CDN edge cache layer in eqiad, all caching servers were accidentally depooled due to insufficient safeguards in tooling (and errors in following the maintenance procedure). Users whose traffic was routed to eqiad were unable to access wikis or any Wikimedia site for about 15 minutes.

Impact

All users of all services behind the eqiad caching layer for about 15 minutes until eqiad was depooled (and that change could propagate).

Estimate we lost about 30k RPS for slightly over 15 minutes: about 27M req lost, or just under 20% of total traffic at the time.

Detection

Both humans and monitoring detected the issue. Humans were slightly faster. Icinga paged and alerted several SREs.

Timeline

All times in UTC.

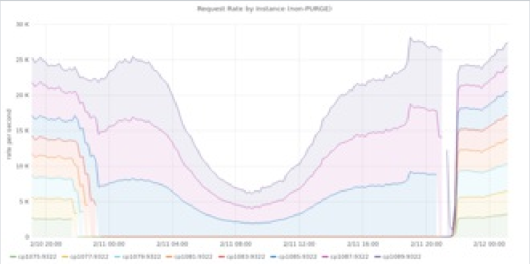
Monday 2020-02-10

- 21:40–23:10: cp1075, cp1077, cp1079, cp1081, cp1083 are depooled for maintenance (BIOS firmware upgrades). The performer of the maintenance, believing that the machines automatically re-pool themselves upon reboot (as they automatically depool themselves prior to shutdown), does not manually re-pool them. **No monitoring or automation informs anyone that over half of eqiad's cache capacity was offline for almost a whole day.**

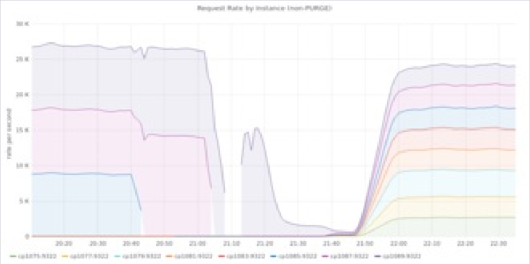
Tuesday 2020-02-11

- 20:40: cp1085 is depooled for maintenance, and not repooled
- 21:02: cp1087, one of two remaining pooled cp-texts in eqiad, is depooled for maintenance **OUTAGE BEGINS**
- 21:04: multiple users on IRC start reporting issues

```
21:04 < Bsadowski1> The wiki is slow
for me
21:04 < musikanimal> everything is
down for me
21:04 < musikanimal> even grafana,
etc.
21:04 < AntiComposite> Very slow
here, some things load, some things
browser timeout
```



Per-ATS-TLS-instance requests/second, timespan including both days



Per-ATS-TLS-instance requests/second, zoomed around the time of the outage

- 21:05: First automated alert from Icinga (non-paging): PROBLEM - restbase endpoints health on restbase1021 is CRITICAL:
- 21:06: First automated page from Icinga: PROBLEM - wiki content on commons #page on commons.wikimedia.org is CRITICAL: CRITICAL - Socket timeout after 10 seconds
- 21:11: eqiad is GeoDNS-depooled
- 21:12: cdanis becomes IC
- 21:17: DNS TTL expires for our load-balancing records; the vast majority of user traffic is no longer routed to eqiad **OUTAGE ENDS**
- 21:21: bblack discovers cause of event, begins determining which hosts can be repooled
- 21:36: all cp servers in eqiad are re-pooled
- 21:36: Icinga recoveries for LVS services start to come in
- 21:45: eqiad is GeoDNS-pooled
- 21:47: incident closed

Conclusions

This outage could have been easily prevented if our infrastructure was more cautious about what it allowed.

What went well?

- automated monitoring quickly detected the incident
- mitigation response (depooling eqiad) was quick and effective
- diagnosis and fix was also quick

What went poorly?

- Insufficient safeguards in confctl/pybal; each let us depool an unlimited number of servers (down to the final one)
  - It is unclear why pybal did not fall back to using servers that weren't marked as pooled.
- SREs and other technical folks whose traffic ingresses at eqiad were unable to use troubleshooting tools like grafana/logstash/phabricator for the duration of the outage.

Where did we get lucky?

- two text CPs in eqiad could handle an impressive amount of load -- no real outage until we were down to just one
- One SRE whose traffic ingresses at eqiad, after receiving Varnish error pages when attempting to use grafana and logstash, took that as a sign to depool eqiad immediately

How many people were involved in the remediation?

- About six SRE for half an hour.

Links to relevant documentation

Where is the documentation that someone responding to this alert should have (runbook, plus supporting docs). If that documentation does not exist, there should be an action item to create it.

Actionables

Explicit next steps to prevent this from happening again as much as possible, with Phabricator tasks linked for every step.

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

- Create an automated alert for 'too many nodes depooled from a service' phab:T245058
- The `depool` & `confctl` commands should print warnings (or error out entirely, unless you override with e.g. a `--force` flag) if too many hosts are depooled from the same service. phab:T245059
- Investigate why Pybal didn't reject a configuration with only one server pooled, and send traffic to some of the depooled servers anyway. phab:T245060
- There should be an easy script for SREs and other technical contributors to override where their own traffic is routed for debugging tools (grafana/logstash/etc). phab:T244761

Category: Incident documentation