



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/20160204-Phabricator

[< Incident documentation](#)

Contents [\[hide\]](#)

- [1 Summary](#)
- [2 Timeline](#)
- [3 Conclusions](#)
- [4 Actionables](#)

Summary

Phabricator (<https://phabricator.wikimedia.org>) was unavailable for 1 hour due to incorrect puppet configuration. The configuration was applied automatically after the server was rebooted during a scheduled maintenance window.

Timeline

- 01:00: Scheduled phabricator maintenance
- 01:00: Moritz reboots iridium (Phabricator host) for a scheduled kernel update.
- 01:04: Phabricator is back up.
- 01:05: Puppet run on iridium updates Phabricator source tree; it had hither to been prevented via a lock file in `/var/run`, which the reboot cleared. The versions of Phabricator and its dependency, libphutil, are now mismatched.
- 01:05: Phabricator is down. Icinga alert from PyBaI: could not depool server about iridium-vcs.eqiad.wmnet!
- 01:08: Mukunda resets Phabricator source tree to desired state
- 01:23: puppet run resets repositories to broken state again but this goes unnoticed because apache is still serving cached good state
- ~01:30: Mukunda ensures the lock file is in `/var/run` as it should be, but without realizing a puppet run happened at 01:24.
- 01:30: Mukunda starts writing patches for puppet to prevent further incidents
- 01:41: Mukunda submits a patch to move the lock to a persistent location:
<https://gerrit.wikimedia.org/r/#/c/268340/>
- 01:44: Mukunda submits a patch to fiix a bug in the git:install puppet module (discovered while writing previous patch): <https://gerrit.wikimedia.org/r/#/c/>
- 01:45: Everything still appears to be running fine and patches submitted. Mukunda finished with maintenance.
 - expectation: Puppet won't touch the repos again unless it's rebooted before <https://gerrit.wikimedia.org/r/#/c/268340/> merges.
 - reality: puppet already reset the repos but opcode cache is serving cached-good-state.
- 06:30: cron.daily logrotate triggers apache graceful restart which clears the opcode cache, reverting once again to the broken state.
- 06:31: Phabricator down again; check_https_phabricator alerts.
- 07:30: Chase checks out release/2015-11-18/1 tag in Phabricator source tree. Phabricator comes back up.
- ...
- Our custom Phabricator "Sprint" extension is not displaying sprint workboards still - [Task T125832](#)
- 16:52: Mukunda restarted apache2 on iridium so that phabricator recognizes sprint.phragile-uri, fixing the Sprint extension
- 16:53: restarted phd to synchronize settings with phabricator

Conclusions

The root cause of this incident is a combination of multiple oversights, the most important are highlighted in bold

[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](#)

below.

1. Puppet is configured to update all phabricator related repositories to a specific hard-coded git tag in the **absence of a lock file**.
 1. The location chosen for the lock file was within `/var/run`
 2. **/var/run is not on persistent storage**, therefore, the lock file is lost when the server is rebooted.
2. Phabricator deployments are being transitioned from a one-off puppet deployment (hard-coded tags in operations/puppet repository) to a scap3-based deployment. (See [Task T114363](#))
3. The deployed state of phabricator has been maintained in the [phabricator/deployment repository](#) for some time, however, puppet hasn't been updated.
4. Normally puppet doesn't update the deployed tags when a lock file is present, however, after a reboot the lock file is lost.
5. Almost immediately after the reboot, the repositories were manually reset to the correct tags
6. **Some time elapsed before the lock file was checked.**
7. Puppet ran during this time without being noticed.
8. **apache graceful restart clears the opcode cache** and apache begins serving the incorrect tags
 1. This is what triggered the outage.
 2. The lock file should have prevented this from occurring, however...
 1. Puppet ran at **01:23** ...Which was after the repo was **fixed at 01:08** ...Unfortunately, that was before **~01:30** when the lock file was confirmed to have been created.
 3. The time it took for Mukunda to double-check the lock file, after repairing the repo, allowed puppet to revert phabricator once again, but without being noticed.

Actionables

- Completely refactor the phabricator module in puppet to remove the automatic git tag pinning behavior. [Task T125851](#)
- move `/srv/phab/repos` to `/srv/repos` then replace `/srv/phab` with a symlink to `/srv/deployment/phabricator/deployment` (Note: Need to schedule a maintenance window for this) [Task T125853](#)
- Use the puppet provider for scap to provision phabricator (<https://gerrit.wikimedia.org/r/#/c/262742/>) -- merged!
- Further deployments of phabricator use scap3. [Task T114363](#) -- done but won't be closed til the above items are resolved

Category: [Incident documentation](#)

This page was last edited on 18 February 2016, at 19:00.

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. See [Terms of Use](#) for details.

[Privacy policy](#) [About](#)
[Wikitech](#)

[Disclaimers](#) [Code of Conduct](#) [Developers](#) [Statistics](#) [Cookie statement](#) [Mobile view](#)

