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/20180111-LegacyEncoding

< Incident documentation

Summary

All non-ascii characters were mangled during edits (including reverts) on Swedish Wiktionary on January 11, 2018.

Analysis

Background: The old Revision::getRevisionText() and the new BlobStore::expandBlob() methods apply the legacy encoding if no flags are provided - the "utf-8" flag is required to bypass this conversion.

As part of the refactoring for MCR, the code for constructing a Revision from an array was consolidated with the code for constructing from a row object. Row objects are required to have the old_flags field set; this field being null or empty would trigger legacy encoding conversion. The same logic was now applied for the 'flags' field when constructing from an array - which was a mistake. No conversion (or indeed decompression or other kinds of decoding) should be applied when constructing from arrays in RevisionStore.php

This mistake led to the legacy encoding conversion to be applied whenever a Revision object was constructed from an array - which is the case whenever a new Revision is prepared for insertion into the database while saving an edit. This caused data corruption by double-encoding.

Solution: Do not apply any processing to the content blob when constructing a Revision from an array (at least not for the normal case of the 'flags' field not being set). Done in https://gerrit.wikimedia.org/r/#/c/403909/

Timeline

- ***2018-01-09***
- 20:12 twentyafterfour@tin: Started scap: Deploy 1.31.0-wmf.16 to test wikis and rebuild l10n. refs T180749
- 20:14 twentyafterfour@tin: scap failed: CalledProcessError Command '/usr/local/bin/mwscript rebuildLocalisationCache.php --wiki="test2wiki" --outdir="/tmp/scap_l10n_3984299293" --threads=10 --lang en --quiet' returned non-zero exit status 1 (duration: 02m 44s)
- 20:21 twentyafterfour@tin: Started scap: Deploy 1.31.0-wmf.16 to test wikis and rebuild l10n. refs T180749 (attempt 2)
- 20:57 twentyafterfour@tin: Finished scap: Deploy 1.31.0-wmf.16 to test wikis and rebuild l10n. refs T180749 (attempt 2) (duration: 36m 34s)
- ***2018-01-10***
- 20:49 twentyafterfour@tin: Synchronized php-1.31.0-wmf.16: Sync wmf.16 to deploy multiple patches from addshore refs T180749 (duration: 10m 23s)
- 21:09 twentyafterfour@tin: Started scap: group0 to 1.31.0-wmf.16 refs T180749
- 21:47 twentyafterfour@tin: Finished scap: group0 to 1.31.0-wmf.16 refs T180749 (duration: 38m 29s)
- 21:53 twentyafterfour@tin: rebuilt and synchronized wikiversions files: group1 wikis to 1.31.0-wmf.16
- **Code pushed out to the 2 wikis with legacy encoding**
  - 21:54 twentyafterfour@tin: Synchronized php: group1 wikis to 1.31.0-wmf.16 (duration: 01m 02s)
- 21:57 twentyafterfour: group1 looks stable. This concludes the MediaWiki train for today.
- ***2018-01-11***

- 17:33 Editors on sv.wiktionary.org reported that all non-ascii characters were mangled on edit <https://phabricator.wikimedia.org/T184749>. The recently deployed MCR patch was suspected as the culprit <https://gerrit.wikimedia.org/r/#/c/399174/>.
- 19:10 Lucas and Daniel investigated. Having a legacy encoding set was suspected as a trigger. But other sites with a legacy encoding set (including sv.wikipedia.org) did not have the problem, so that idea was discarded. As it turns out, we failed to make the connection that sv.wikipedia and the other sites with legacy encoding set didn't have the new code yet.
- 20:12 **Deployment was rolled back, train halted.** (Live for roughly 24 hours)
  - 20:12 twentyafterfour@tin: rebuilt and synchronized wikiversions files: Rollback group1 to wmf.15 due to phab:T184749 refs phab:T180749
- 20:31 Adam realized that the legacy encoding was the trigger after all, reproduced the error locally, and found the bug. phab:T184749#3894944
- ***2018-01-12***
- 15:07 Adam and Daniel worked together on patches that fix the behavior and provide comprehensive regression tests.
  - RevisionStore, fix loadSlotContent with no $blobFlags https://gerrit.wikimedia.org/r/403909
  - Add tests for legacy encoding when constructing RevisionRecords - https://gerrit.wikimedia.org/r/403926
  - Extra tests for SqlBlobStore with 'windows-1252' legacy encoding - https://gerrit.wikimedia.org/r/#/c/403754/
  - selenium, update page spec to include more chars - https://gerrit.wikimedia.org/r/#/c/404075/

Conclusions

- Code author and reviewers failed to spot the issue; This may be explained by the fact that it is counter-intuitive that passing no encoding flags would cause legacy encoding conversion, and a flag ("utf-8") needs to be passed to prevent conversion. This was not documented on the old Revision.php class. Also, the intended default behavior differs between construction of a Revision from a database row (here, conversion needs to be applied per default) and construction from a programmatically constructed array (here, the expectation is that no conversion is applied). Author and reviewers failed to take into account this difference in expected behavior when refactoring the Revision class.
- No unit tests existed for the expected behavior of the Revision class with respect to applying a legacy encoding. Such unit tests would have prevented the oversight described above. It should be noted that an effort was made to substantially improve test coverage of Revision prior to the refactoring (T180210), but the edge case in question was overlooked due to a lack of explicit documentation.
- Selenium tests run on submit by Jenkins did not spot a problem, because the relevant tests case did not cover non-ascii characters (this has since been fixed).
- No problems were spotted on the beta cluster, even though the beta wiki is configured with a legacy encoding. This is because we don't run any automated tests for core against the beta cluster.
- No problems were observed on test.wikipedia.org, because that wiki does not have a legacy encoding configured, see InitialiseSettings.php.

Actionables

Concrete actionables:

- Make sure we automatically run browser tests that test editing with non-ascii characters (in content and title) on a wiki with legacy encoding configured. (Done in gerrit:404075)
- Run daily Jenkins job targeting beta cluster. Tracked as phab:T185011.

Recommended best practices:

- More eyes on the code! More reviewers with different perspectives may have helped to spot the issue.
- More documentation for edge cases! The need for arcane knowledge leads to bugs.
- More unit tests! When behavior depends on global state (as is the case with the legacy encoding), different values for the relevant global variables need to be tested.

Category: Incident documentation