I lately realized that upstream valgrind is not properly working on arm64 and therefore on openSUSE Leap 15. It would be good if we could run a test of valgrind on openQA to fetch this regressions early.

Suggestion

Implement a simple check for valgrind -q --leak-check=full --show-reachable=yes $(which true) as a package self-test comparable to https://build.opensuse.org/package/view_file/openSUSE:Factory/widelands/widelands.spec?expand=1, line 133

In my case the problem is a missing instruction decoding which you hit with newer kernel and newer glibc:
https://bugzilla.suse.com/show_bug.cgi?id=1086543

I can think of a simple valgrind -q --leak-check=full --show-reachable=yes $(which true) test as well as one executed on a faulty binary if that is necessary. WDYT?

Sounds like a good idea. Do you have any further references regarding the "is not properly working". Do you have mailing list discussions or bug reports which describe what exactly did not work?

I don't know if we want to do a functional test, like if it finds the errors in a faulty binaries. I think there are many corner cases and we can easily miss some cases.

I can think of a simple valgrind -q --leak-check=full --show-reachable=yes $(which true) test as well as one executed on a faulty binary if that is necessary. WDYT?

Also I wonder about upstream test suites which we could also execute during build time in OBS as well as a potential post-install test, still executed in OBS, as I assume valgrind is neither about GUI testing nor system level integration tests. WDYT?
valgrind -v --trace-children=yes firefox
valgrind -v --trace-children=yes soffice

I think we can add the test suite in the OBS build, but I suppose that won't be enough.

#3 - 2018-06-14 18:49 - okurz
mbrugger wrote:

    I can think of a simple valgrind -q --leak-check=full --show-reachable=yes $(which true) test as well as one executed on a faulty binary if
    that is necessary. WDYT?

    I don't know if we want to do a functional test, like if it finds the errors in a faulty binaries. I think there are many corner cases and we can easily
    miss some cases.

Sorry but I don't understand now, what do you suggest we should test in openQA?

    valgrind -v --trace-children=yes firefox
    valgrind -v --trace-children=yes soffice
    I think we can add the test suite in the OBS build, but I suppose that won't be enough.

Ok, why would that not be enough? What else would you execute then?

#4 - 2018-06-15 19:08 - okurz
- Target version changed from future to future

#5 - 2018-11-06 11:23 - okurz
- Status changed from New to Feedback
- Target version changed from future to Milestone 25

Hi mbrugger, could you help me answer my last two questions in #36751#note-3?

#6 - 2018-11-07 10:48 - mbrugger
okurz wrote:

    Hi mbrugger, could you help me answer my last two questions in #36751#note-3?

After a chat with Dirk I think the best would be to add a simple check as you describe in #36751#note-1
Although it will only help us to find a totally broken valgrind as we had in aarch64.
Unfortunately the valgrind testsuite is really fragile and most probably will throw a lot of false positives.

#7 - 2018-11-07 10:54 - okurz
- Project changed from openQA Tests to QA
- Subject changed from [functional][u] test valgrind with openQA to [functional][u] test valgrind
- Description updated
- Category deleted (New test)
- Status changed from Feedback to Workable
- Assignee deleted (okurz)
- Target version changed from Milestone 25 to Milestone 22

yes, that makes sense, thank you. Let's try with a package based check. I do not see the need to do that within a VM within openQA for now.

#8 - 2018-11-23 08:59 - dirkmueller
Added a selftest for valgrind in the %check section as part of the 3.14 update.

#9 - 2018-12-05 10:00 - zluo
- Status changed from Workable to In Progress
- Assignee set to zluo
take over

#10 - 2018-12-05 11:30 - okurz
zluo please keep in mind that this is not a request for an openQA test but still should not be too hard

#11 - 2018-12-05 13:58 - zluo
- Assignee deleted (zluo)

after talked with mgriessmeier, I don't thin I won't be able do this manual testing regarding to workflow with OBS, spec changes etc. But mkittler is willing to try this...

#12 - 2018-12-05 14:06 - zluo
I'll open a ticket about valgrind test module for openQA.

#13 - 2018-12-05 14:10 - mkittler
- Status changed from In Progress to Resolved

The package already contains a "patent pending" self-test: https://build.opensuse.org/package/view_file/devel-tools/valgrind/valgrind.spec?expand=1

This test is also present in the openSUSE:Factory version. The package containing the test also builds for aarch64. So it looks like the ticket has been resolved with https://build.opensuse.org/request/show/652030. If that's not true, just reopen the ticket. However, then I'm not sure what is missing here.

#14 - 2018-12-05 15:26 - okurz
- Assignee set to mkittler

mkittler what a nice coincidence! Yep, that should be enough, cool!

zluo wrote:

I'll open a ticket about valgrind test module for openQA.

Please don't. Why should we need that?