New issue

Jump to bottom

Use-after-free when opening a new document #645

Closed **hfiguiere** opened this issue on Mar 7, 2021 · 6 comments

Labels bug

hfiguiere commented on Mar 7, 2021 Contributor

Describe the bug
This only happen if LeoCAD reopen the last document.

This need leocad compiled with address sanitizer. Rough patch:

```
diff --git a/leocad.pro b/leocad.pro
index 5b9abb49..39dcf87f 100644
--- a/leocad.pro
+++ b/leocad.pro
@@ -18,6 +18,11 @@ qtHaveModule(gamepad) {
 INCLUDEPATH += qt common
 CONFIG += precompile_header incremental c++11 force_debug_info

+linux-* {
+        QMAKE_CXXFLAGS += -fsanitize=address
+        LIBS += -fsanitize=address
+}
+
 win32 {
        RC_ICONS = resources/leocad.ico
        VERSION = 21.1.0.0
```

To Reproduce
Steps to reproduce the behavior:

1. Start leocad with the last document reopened
2. File > New
3. Crash

Expected behavior
Shouldn't crash

Stack trace

```
=================================================================
==128700==ERROR: AddressSanitizer: heap-use-after-free on address 0x6120003ef6e0 at pc 0x000000760c51 bp 0x7ffc67971560 sp 0x7ffc67971558
READ of size 8 at 0x6120003ef6e0 thread T0
    #0 0x760c50 in std::vector<Image, std::allocator<Image> >::operator[](unsigned long) /usr/include/c++/10/bits/stl_vector.h:1046
    #1 0x760c50 in lcTexture::Upload(lcContext*) common/lc_texture.cpp:207
    #2 0x59c83e in lcPiecesLibrary::UploadTextures(lcContext*) common/lc_library.cpp:1555
    #3 0x6fb344 in lcScene::Draw(lcContext*) const common/lc_scene.cpp:401
    #4 0x7afb9a in lcView::OnDraw() common/lc_view.cpp:972
    #5 0x7f0114c6c08c  (/lib64/libQt5Widgets.so.5+0x20d08c)
    #6 0x7f0114c4db1d in QWidget::event(QEvent*) (/lib64/libQt5Widgets.so.5+0x1eeb1d)
    #7 0x7f0114c0cec2 in QApplicationPrivate::notify_helper(QObject*, QEvent*) (/lib64/libQt5Widgets.so.5+0x1adec2)
    #8 0x7f0113f8fbd7 in QCoreApplication::notifyInternal2(QObject*, QEvent*) (/lib64/libQt5Core.so.5+0x281bd7)
    #9 0x7f0114c45be9 in QWidgetPrivate::sendPaintEvent(QRegion const&) (/lib64/libQt5Widgets.so.5+0x1e6be9)
    #10 0x7f0114c6beee in QOpenGLWidget::resizeEvent(QResizeEvent*) (/lib64/libQt5Widgets.so.5+0x20ceee)
    #11 0x7f0114c4e573 in QWidget::event(QEvent*) (/lib64/libQt5Widgets.so.5+0x1ef573)
    #12 0x7f0114c0cec2 in QApplicationPrivate::notify_helper(QObject*, QEvent*) (/lib64/libQt5Widgets.so.5+0x1adec2)
    #13 0x7f0113f8fbd7 in QCoreApplication::notifyInternal2(QObject*, QEvent*) (/lib64/libQt5Core.so.5+0x281bd7)
    #14 0x7f0114c45b0d in QWidgetPrivate::sendPendingMoveAndResizeEvents(bool, bool) (/lib64/libQt5Widgets.so.5+0x1e6b0d)
    #15 0x7f0114c4a496 in QWidgetPrivate::show_helper() (/lib64/libQt5Widgets.so.5+0x1eb496)
    #16 0x7f0114c4d652 in QWidgetPrivate::setVisible(bool) (/lib64/libQt5Widgets.so.5+0x1ee652)
    #17 0x5e3c39 in lcMainWindow::SetCurrentModelTab(lcModel*) common/lc_mainwindow.cpp:1580
    #18 0x465632 in Project::SetActiveModel(int) common/project.cpp:155
    #19 0x54d8f9 in lcApplication::SetProject(Project*) common/lc_application.cpp:269
    #20 0x5eec61 in lcMainWindow::NewProject() common/lc_mainwindow.cpp:2293
    #21 0x5fbec7 in lcMainWindow::HandleCommand(lcCommandId) common/lc_mainwindow.cpp:2545
    #22 0x7f0113fbf3bf  (/lib64/libQt5Core.so.5+0x2b13bf)
    #23 0x7f0114c06645 in QAction::triggered(bool) (/lib64/libQt5Widgets.so.5+0x1a7645)
    #24 0x7f0114c08f30 in QAction::activate(QAction::ActionEvent) (/lib64/libQt5Widgets.so.5+0x1a9f30)
    #25 0x7f0114c09b0e in QAction::event(QEvent*) (/lib64/libQt5Widgets.so.5+0x1aab0e)
    #26 0x7f0114c0cec2 in QApplicationPrivate::notify_helper(QObject*, QEvent*) (/lib64/libQt5Widgets.so.5+0x1adec2)
    #27 0x7f0113f8fbd7 in QCoreApplication::notifyInternal2(QObject*, QEvent*) (/lib64/libQt5Core.so.5+0x281bd7)
    #28 0x7f01145dbbc1 in QShortcutMap::dispatchEvent(QKeyEvent*) (/lib64/libQt5Gui.so.5+0x190bc1)
    #29 0x7f01145dc112 in QShortcutMap::tryShortcut(QKeyEvent*) (/lib64/libQt5Gui.so.5+0x191112)
    #30 0x7f0114597cc6 in QWindowSystemInterface::handleShortcutEvent(QWindow*, unsigned long, int, QFlags<Qt::KeyboardModifier>, unsigned int, unsigned int, unsigned int, QString const&, bool, unsigned short) (/lib64/libQt5Gui.so.5+0x14ccc6)
    #31 0x7f01145af85e in QGuiApplicationPrivate::processKeyEvent(QWindowSystemInterfacePrivate::KeyEvent*) (/lib64/libQt5Gui.so.5+0x16485e)
    #32 0x7f01145948cb in QWindowSystemInterface::sendWindowSystemEvents(QFlags<QEventLoop::ProcessEventsFlag>) (/lib64/libQt5Gui.so.5+0x1498cb)
    #33 0x7f00ffcfe47d  (/lib64/libQt5XcbQpa.so.5+0x6b47d)
    #34 0x7f0112b38a9e in g_main_context_dispatch (/lib64/libglib-2.0.so.0+0x53a9e)
    #35 0x7f0112b8aa97  (/lib64/libglib-2.0.so.0+0xa5a97)
    #36 0x7f0112b35e72 in g_main_context_iteration (/lib64/libglib-2.0.so.0+0x50e72)
    #37 0x7f0113fdc6f2 in QEventDispatcherGlib::processEvents(QFlags<QEventLoop::ProcessEventsFlag>) (/lib64/libQt5Core.so.5+0x2ce6f2)
    #38 0x7f0113f8e57a in QEventLoop::exec(QFlags<QEventLoop::ProcessEventsFlag>) (/lib64/libQt5Core.so.5+0x28057a)
    #39 0x7f0113f961b3 in QCoreApplication::exec() (/lib64/libQt5Core.so.5+0x2881b3)
    #40 0x45b78d in main qt/qtmain.cpp:217
    #41 0x7f01137771e1 in __libc_start_main (/lib64/libc.so.6+0x281e1)
    #42 0x45c56d in _start (/home/hub/source/leocad/build/release/leocad+0x45c56d)

0x6120003ef6e0 is located 288 bytes inside of 320-byte region [0x6120003ef5c0,0x6120003ef700)
freed by thread T0 here:
    #0 0x7f01152e75b7 in operator delete(void*) (/lib64/libasan.so.6+0xad5b7)
    #1 0x775bb7 in lcView::DestroyResources(lcContext*) common/lc_view.cpp:342

previously allocated by thread T0 here:
    #0 0x7f01152e6bb7 in operator new(unsigned long) (/lib64/libasan.so.6+0xacbb7)
    #1 0x775b33 in lcView::CreateResources(lcContext*) common/lc_view.cpp:188

SUMMARY: AddressSanitizer: heap-use-after-free /usr/include/c++/10/bits/stl_vector.h:1046 in std::vector<Image, std::allocator<Image> >::operator[](unsigned long)
Shadow bytes around the buggy address:
  0x0c2480075e80: fa fa fa fa fa fa fa fa fd fd fd fd fd fd fd fd
  0x0c2480075e90: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c2480075ea0: fd fd fd fd fd fd fd fd fd fd fd fd fa fa fa fa
  0x0c2480075eb0: fa fa fa fa fa fa fa fa fd fd fd fd fd fd fd fd
  0x0c2480075ec0: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
=>0x0c2480075ed0: fd fd fd fd fd fd fd fd fd fd fd fd[fd]fd fd fd
  0x0c2480075ee0: fa fa fa fa fa fa fa fa fd fd fd fd fd fd fd fd
  0x0c2480075ef0: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c2480075f00: fd fd fd fd fd fd fd fd fd fd fd fd fa fa fa fa
  0x0c2480075f10: fa fa fa fa fa fa fa fa fd fd fd fd fd fd fd fd
  0x0c2480075f20: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
Shadow byte legend (one shadow byte represents 8 application bytes):
  Addressable:           00
  Partially addressable: 01 02 03 04 05 06 07
  Heap left redzone:       fa
  Freed heap region:       fd
  Stack left redzone:      f1
  Stack mid redzone:       f2
  Stack right redzone:     f3
  Stack after return:      f5
  Stack use after scope:   f8
  Global redzone:          f9
  Global init order:       f6
  Poisoned by user:        f7
  Container overflow:      fc
  Array cookie:            ac
  Intra object redzone:    bb
  ASan internal:           fe
  Left alloca redzone:     ca
  Right alloca redzone:    cb
  Shadow gap:              cc
==128700==ABORTING
```

Version (please complete the following information):

- OS: Linux Fedora 33
- LeoCAD Version: git `master` commit `de3a3ad`

hfiguiere added the bug label on Mar 7, 2021

leozide commented on Mar 8, 2021 Owner

I can't repro this, some steps would be nice.

hfiguiere commented on Mar 8, 2021 Contributor Author

Apparently it needs a specific file to be reopened at startup.

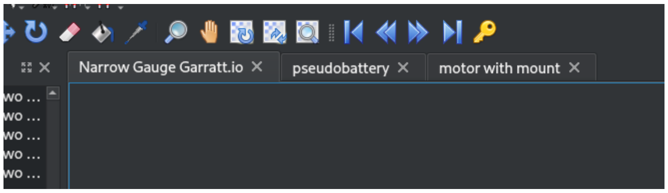
1. Open the attached file (inside the .zip) issue-645.zip and make sure Leocad reopen the last file at startup.
2. Quit Leocad
3. Start Leocad
4. File > New
5. it crashes.

leozide commented on Mar 9, 2021 Owner

Doesn't happen, probably has something to do with your tab layout. Can you describe your tabs?

hfiguiere commented on Mar 9, 2021 Contributor Author

Indeed it seems it's specific to which tabs are open. With this I can reproduce:



Given that I couldn't reproduce with just "motor with mount" either.

leozide closed this as completed in `233affe` on Mar 11, 2021

carnil commented on Apr 26, 2021

There seems to be a CVE id assigned to this issue, CVE-2021-31804

trevorsandy commented on Apr 26, 2021 Contributor

This vulnerability is corrected in v21 03. So the recommended action would be to update your installation to at least this version.

Cheers,

Assignees

No one assigned

Labels

bug

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

4 participants