

QGIS Application - Bug report #19993

QGIS Crashed running Python code obtained form StackExchnge

2018-10-02 11:34 AM - Thomas Kibbey

Status:	Rejected	
Priority:	Normal	
Assignee:	Thomas Kibbey	
Category:	PyQGIS Console	
Affected QGIS version:	3.2.1	Regression?: No
Operating System:	64 bit, 32GB Windows 10	Easy fix?: No
Pull Request or Patch supplied:	No	Resolution: invalid
Crashes QGIS or corrupts data:	Yes	Copied to github as #: 27815

Description

This was the code that caused the crash:

```
import itertools
layer = iface.activeLayer()
feats = [ feat for feat in layer.getFeatures() ]
selected_feats = []
selected_feats.append(feats[0])
del feats[0]
k=0
while len(feats) !=0:
    for i, feat in enumerate(feats):
        d = selected_feats[k].geometry().distance(feat.geometry())
        if d == 0:
            selected_feats.append(feat)
            k +=1
            del feats[i]
lines = [ feat.geometry().asPolyline() for feat in selected_feats ]
epsg = layer.crs().postgisSrid()
uri = "LineString?crs=epsg:" + str(epsg) + "&field=id:integer""&index=yes"
mem_layer = QgsVectorLayer(uri,
    '4testing_feats_ordered',
    'memory')
prov = mem_layer.dataProvider()
feats = [ QgsFeature() for i in range(len(selected_feats)) ]
for i, feat in enumerate(feats):
    feat.setAttributes([i])
    feat.setGeometry(QgsGeometry.fromPolyline(lines[i]))
prov.addFeatures(feats)
QgsProject.instance().addMapLayer(mem_layer)
```

This was the bug report after the crash:

User Feedback

Report Details

Crash ID: e75beb8f185e848f989ceaa0bae49ce173157d10

Stack Trace

SpatialIndex::MVRTree::Data::storeToByteArray :
 SpatialIndex::RTree::Data::getShape :
 SpatialIndex::RTree::Data::getShape :
 QgsSpatialIndex::insertFeature :
 QgsSpatialIndex::insertFeature :
 QgsProcessingModelOutput::toVariant :
 std::_String_alloc<std::_String_base_types<char,std::allocator<char> > >::_Geta1 :
 PyCFunction_FastCallDict :
 PyObject_GenericGetAttr :
 PyEval_EvalFrameDefault :
 PyErr_Occurred :
 PyEval_EvalCode :
 PyDict_SetItemId :
 PyDict_SetItemId :
 PyCFunction_FastCallDict :
 PyObject_GenericGetAttr :
 PyEval_EvalFrameDefault :
 PyObject_GenericGetAttr :
 PyEval_EvalFrameDefault :
 PyErr_Occurred :
 PyObject_GenericGetAttr :
 PyEval_EvalFrameDefault :
 PyErr_Occurred :
 PyObject_GenericGetAttr :
 PyEval_EvalFrameDefault :
 PyObject_GenericGetAttr :
 PyEval_EvalFrameDefault :
 PyObject_GenericGetAttr :
 PyEval_EvalFrameDefault :
 PyFunction_FastCallDict :
 PyObject_CallFunctionObjArgs :
 PyObject_Call :
 PyInit_sip :
 PyInit_Qsci :
 QWidget::event :
 QFrame::event :
 QAbstractScrollArea::event :
 PyInit_Qsci :
 QApplicationPrivate::notify_helper :
 QApplication::notify :
 QgsApplication::notify :
 QCoreApplication::notifyInternal2 :
 QSizePolicy::QSizePolicy :
 QSizePolicy::QSizePolicy :
 QApplicationPrivate::notify_helper :
 QApplication::notify :
 QgsApplication::notify :
 QCoreApplication::notifyInternal2 :
 QGuiApplicationPrivate::processKeyEvent :
 QWindowSystemInterface::sendWindowSystemEvents :
 QEventDispatcherWin32::processEvents :
 CallWindowProcW :
 DispatchMessageW :
 QEventDispatcherWin32::processEvents :
 qt_plugin_query_metadata :
 QEventLoop::exec :
 QCoreApplication::exec :
 main :
 BaseThreadInitThunk :
 RtlUserThreadStart :

QGIS Info

QGIS Version: 3.2.1-Bonn

QGIS code revision: commit:1edf372fb8

Compiled against Qt: 5.9.2
Running against Qt: 5.9.2
Compiled against GDAL: 2.2.4
Running against GDAL: 2.2.4

System Info

CPU Type: x86_64
Kernel Type: winnt
Kernel Version: 10.0.17134

History

#1 - 2018-10-03 06:08 PM - Alessandro Pasotti

- *Resolution set to invalid*
- *Status changed from Open to Rejected*

I'm afraid there is nothing we can do, QGIS C++ is wrapped by SIP binding generator in such a way that you can easily crash QGIS but you get the power and the speed you need for complex tasks.

If you want to know more about this: http://enki-editor.org/2014/08/23/Pyqt_mem_mgmt.html

#2 - 2019-01-07 05:29 PM - Jürgen Fischer

- *Description updated*