<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
<th>Start date:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
<td>Due date:</td>
<td></td>
</tr>
<tr>
<td>Assignee:</td>
<td>Marko Lindqvist</td>
<td>% Done:</td>
<td>0%</td>
</tr>
<tr>
<td>Category:</td>
<td>Client</td>
<td>Estimated time:</td>
<td>0.00 hour</td>
</tr>
<tr>
<td>Sprint/Milestone:</td>
<td>2.6.2.1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Description**

I ran 'freeciv-qt -t augmentedi'. I was starting games with random rulesets and random changes to Game options, and if the game started ok, then quit and try again. I was looking for trouble (like a core dump), found it, then got confused with my own note-taking and can't reproduce it, or be sure what I did when it triggered.

Core was generated by 'freeciv-qt'.

Program terminated with signal SIGSEGV, Segmentation fault.

```
#0  0x00007f31eb0a5ee4 in QActionPrivate::sendDataChanged() () from /usr/lib/x86_64-linux-gnu/libQt5Widgets.so.5
   [Current thread is 1 (Thread 0x7f31e53c74c0 (LWP 3553034))]
(gdb) bt full
#0  0x00007f31eb0a5ee4 in QActionPrivate::sendDataChanged() () at /usr/lib/x86_64-linux-gnu/libQt5Widgets.so.5
#1  0x0000563b00347976 in gov_menu::update() (this=0x563b02558380) at /usr/include/x86_64-linux-gnu/qt5/QtCore/qarraydata.h:211
gov = 0x563b05206ab0
revel_gov = 0x563b05206480
sprite = <optimized out>
gov_count = 9
i = 6
j = 5
#2  0x0000563b00347b84 in gov_menu::update_all() () at menu.cpp:735
m = <optimized out>
    container = {{c = {{q_hash = {{d = 0x563b01d07a0, e = 0x563b01d07a0}}}, i = {i = {i = 0x563b01d07a0}, control = 1}}},
#3  0x0000563b00353089 in real_menus_update () at menu.cpp:580
#4  real_menus_update() () at menu.cpp:571
#5  0x0000563b003530be in qt_start_turn() () at menu.cpp:77
#6  0x0000563b00340601 in fc_client::server_input(int) (this=0x563b017f8270, sock=41) at fc_client.cpp:462
#7  0x0000563b0039689f in client_packet_input (packet=packet@entry=0x563b04ed9010, type=128) at client.c:416
FUNCTION = "client_packet_input"
#9  0x0000563b0039d79c in input_from_server (fd=fd@entry=41) at clinet.c:416
FUNCTION = "input_from_server"
```

2021-08-08
```
#24 0x00000563b00445ee1 in fc_client::fc_main(QApplication*) (this=0x563b017f8270, qapp=0x563b0143c120) at fc_client.cpp:257
#25 0x00000563b003226d3 in qtg_ui_main(int, char**) (argc=<optimized out>, argv=<optimized out>) at gui_main.cpp:191
qpm = <optimized out>
app_icon = (d = 0x563b01e2dc70)
#26 0x00000563b00396fce in client_main (argc=1, argv=0x7fff82e71708) at client_main.c:685
i = 1
loglevel = LOG_NORMAL
ui_options = <optimized out>
option = <optimized out>
fatal_assertions = -1
--Type <RET> for more, q to quit, c to continue without paging--c
aii = 1
FUNCTION = "client_main"
#27 0x000005631ee9fa1e3 in __libc_start_main (main=0x563b00320460 <main(int, char**)>, argc=1, argv=0x7fff82e71708, init=<optimized out>, fini=<optimized out>, rtld_fini=<optimized out>, stack_end=0x7fff82e716f8) at ../csu/libc-start.c:308
result = <optimized out>
unwind_buf = {cancel_jmp_buf = {{jmp_buf = {0, 4538260550429502599, 94811406342016, 140735389570816, 0, 0, 7886408311432866951, 7849482102057397383}, mask_was_saved = 0}}, priv = {pad = {0x0, 0x0, 0x7fff82e71718, 0x7f31ec77c190}, data = {prev = 0x0, cleanup = 0x0, canceltype = -209874488}}} not_first_call = <optimized out>
#28 0x00000563b003217ae in _start () at gui_main.cpp:114

Related issues:
| Related to Freeciv - Bug #857869: Qt SEGV in Cancel of Load Game (2.6.1+) | Closed |
| Related to Freeciv - Bug #858251: SEGv in name_translation_get (2.6.1+, qt) | Closed |

History

#1 - 2020-01-30 04:31 PM - Marko Lindqvist
Chippo Elder wrote:

```
#5 0x00000563b003580be in qt_start_turn() () at menu.cpp:77
#6 0x00000563b00340601 in qtg_update_timeout_label() () at mapview.cpp:1086
```

This call-chain is showing up in several of your backtraces in different tickets. I think qt_start_turn() gets called when client state is not suitable for it. I don't know why Qt-client has that qt_start_turn() call in qtg_update_timeout_label(). Other clients do not have equivalent call in their update_timeout_label() function.

#2 - 2020-02-01 01:01 PM - Jacob Nevins
- Related to Bug #857869: Qt SEGV in Cancel of Load Game (2.6.1+) added

#3 - 2020-02-10 11:40 PM - Jacob Nevins
- Related to Bug #858251: SEGv in name_translation_get (2.6.1+, qt) added

#4 - 2020-04-22 01:00 PM - Marko Lindqvist
- Category set to Client
- Status changed from New to In Progress
- Sprint/Milestone set to 2.6.3

Marko Lindqvist wrote:

I don't know why Qt-client has that qt_start_turn() call in qtg_update_timeout_label(). Other clients do not have equivalent call in their update_timeout_label() function.

I considered patching over that design in S2_6, but it turned out that there's no low-impact way to do it, so I'm going to full clean gui API change even in S2_6. As qt_start_turn() call goes away from qtg_update_timeout_label(), any ticket with backtrace with that call chain should be closed if my patch gets accepted.

#5 - 2020-04-22 02:15 PM - Marko Lindqvist
- File 0022-Fix-Qt-client-crash-on-timeout-label-update.patch added
- File 0010-Fix-Qt-client-crash-on-timeout-label-update.patch added
- File 0006-Fix-Qt-client-crash-on-timeout-label-update.patch added

2021-08-08 2/3
Status changed from In Progress to Resolved

Patches that reimplement affected code. In any case this is cleaner implementation. I believe it to also avoid the crashes of the old implementation.

#6 - 2020-04-25 08:20 PM - Marko Lindqvist
- Status changed from Resolved to Closed
- Assignee set to Marko Lindqvist

#7 - 2020-11-16 02:10 AM - Marko Lindqvist
- Sprint/Milestone changed from 2.6.3 to 2.6.2.1

Files
<table>
<thead>
<tr>
<th>Patch Name</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>0022-Fix-Qt-client-crash-on-timeout-label-update.patch</td>
<td>10.9 KB</td>
<td>2020-04-22</td>
<td>Marko Lindqvist</td>
</tr>
<tr>
<td>0010-Fix-Qt-client-crash-on-timeout-label-update.patch</td>
<td>11.3 KB</td>
<td>2020-04-22</td>
<td>Marko Lindqvist</td>
</tr>
<tr>
<td>0006-Fix-Qt-client-crash-on-timeout-label-update.patch</td>
<td>13.4 KB</td>
<td>2020-04-22</td>
<td>Marko Lindqvist</td>
</tr>
</tbody>
</table>