In fill_unit_type_sprite_array() [tilespec.c::4150]: assertion 'uspr != NULL' failed, plus a SEGV (2.6.1+, civ2civ3_earth)

2020-01-31 07:43 AM - Chippo Elder

Status: Closed
Priority: Normal
Assignee: Marko Lindqvist
Category: Client
Sprint/Milestone: 2.6.2

Description
I used freeciv-mp-gtk3 (2.6.1+) to install the civ2civ3_earth modpack. When starting a game with that ruleset, as you press Start, you hit this assertion and then a SEGV.

I'll test reproducibility of the assertion failure and if it's reproducible, I'll run with -F and get a backtrace of the assert failure. In this edit is the backtrace of the SEGV.

1: in fill_unit_type_sprite_array() [tilespec.c::4150]: assertion 'uspr != NULL' failed.
1: Please report this message at https://www.hostedredmine.com/projects/freeciv

Thread 1 "freeciv-qt" received signal SIGSEGV, Segmentation fault.
0x00007ffff7547424 in QPixmap::width() const () from /usr/lib/x86_64-linux-gnu/libQt5Gui.so.5
(gdb) bt full
#0  0x00007ffff7547424 in QPixmap::width() const () at /usr/lib/x86_64-linux-gnu/libQt5Gui.so.5
#1  0x0000555555635579 in qtg_get_sprite_dimensions(sprite*, int*, int*) (sprite=0x7ffff6ab54c0, width=0x7fffffffa480, height=0x7fffffffa484) at sprite.cpp:164
#2  0x00005555556a158b in qtg_canvas_put_sprite_full(canvas*, int, int, sprite*) (pcanvas=0x55555b435470, canvas_x=0, canvas_y=0, sprite=0x7ffff6ab54c0) at canvas.cpp:151
width = -21984
height = 32767

1302 i = <optimized out>
1332 tile_sprs =              {{foggable = false, sprite = 0x7ffff6ab54c0, offset_x = 0, offset_y = 0}, {foggable = false, sprite = 0x0, offset_x = 0, offset_y = 0}, {foggable = false, sprite = 0x55555b0fa550, offset_x = 0, offset_y = 0}, {foggable = 151, sprite = 0x7ffff739a988, offset_x = 0, offset_y = 0}, {foggable = true, sprite = 0x7ffff70dca9a <QCoreApplication::self>, offset_x = 0, offset_y = 0}, {foggable = 64, sprite = 0x7fffffffa640, offset_x = -23121, offset_y = 32767}, {foggable = false, sprite = 0x0, offset_x = 124, offset_y = 119}, {foggable = 80, sprite = 0x7ffff6e273c0 <QAction::staticMetaObject>, offset_x = -222130, offset_y = 1600}, {foggable = 76, sprite = 0x7fffffffa960, offset_x = -157955616, offset_y = 32767}, {foggable = 96, sprite = 0x7ffff69cf995, offset_x = -22168, offset_y = 32767}, {foggable = false, sprite = 0x55555c3b3e60, offset_x = 1547386464, offset_y = 2021-08-09 1/9
I got a backtrace for the assertion failure (see below). To trigger it, I need to play at least one turn of Ancients (another ruleset), leave game and then try to start a civ2civ3_earth game. If you just charge straight into a civ2civ3_earth game, everything goes correctly.

So far, both backtraces have come from the qt client, but I didn't put it into the Subject yet, 'cos at least one of them might be a server-side problem.

After a bit more research and some tests with the gk client, an admin can help me break this ticket into as many separate tickets as is appropriate.

0: in fill_unit_type_sprite_array() [tilespec.c::4150]: assertion 'uspr != NULL' failed.
0: Please report this message at https://www.hostedredmine.com/projects/freeciv

---Type <RET> for more, q to quit, c to continue without paging---
Thread 1 'freeciv-qt' received signal SIGABRT, Aborted.
raise (sig=<optimized out>) at ../sysdeps/unix/sysv/linux/raise.c:50
50  ../sysdeps/unix/sysv/linux/raise.c:50
50  ../sysdeps/unix/sysv/linux/raise.c:50
50  no such file or directory

Result = <optimized out>

unwind_buf = {cancel_jmp_buf = {{jmp_buf = {0, 340965609587617117, 93824992692096, 140737488337904, 0, 0, 613741054480996223, 6137392473419049711}, mask_was_saved = 0}}, priv = {pad = {0x0, 0x0, 0x7fffffffbc08, 0x7ffff7ffe190}, data = {prev = 0x0, cleanup = 0x0, canceltype = -17400))

not_first_call = <optimized out>

#27 0x000055555555c47ae in _start () at gui_main.c::114
#2 0x00005555566665b7eb in fill_unit_type_sprite_array
t={@entry=0x55555566670, sprs=sprs@entry=0x7fffffffa52, putype=putype@entry=0x5555556c8e40, pctype=0x0, pcw=0, pcolor=0x0, pcity=0x0, putype=0x0) at tilespec.c:4154
usrp = 0x0
FUNCTION = "fill_unit_type_sprite_array"

#3 0x00005555566665b7eb in fill_unit_type_sprite_array (backdrop=optimized out), stack=failed, punitsize=0x5555556c8e40, sprs=0x7fffffffa52, t={@entry=0x55555566670) at tilespec.c:4180
save_sprs = 0x7fffffffa10
iptp = {optimized out}
backdrop = {optimized out}
tileno = {optimized out}
dir = {optimized out}
texts = {optimized out}

#4 fill_unit_sprite_array (t={optimized out}, sprs@sprtptr=0x7fffffffa510, layer=layer@entry=LAYAYR_FOCUS_UNIT, ptile=optimized out, pti=entry=0x0, pedge=pedge@entry=0x0, pcorner=pcorner@entry=0x0, punit=punit@entry=0x0, pct=0x0, pcy=0, ptye=0x0, ptye=0x0) at mapview_common.c:1325
do_draw_unit = <optimized out>
owner = 0x0
do_draw_unit = {optimized out}
solid_bg = {optimized out}
FUNCTION = "fill_unit_sprite_array"

#5 0x00005555566665b7eb in put_one_element (pcanvas=pcanvas@entry=0x5555556c3680, zoom=zoom@entry=1, layer=layer@entry=LAYAYR_FOCUS_UNIT, ptile=ptile@entry=0x0, pedge=pedge@entry=0x0, pcorner=pcorner@entry=0x0, punit=punit@entry=0x0, pct=0x0, pcy=0, ptyes=0x0, putype=putype@entry=0x0) at mapview_common.c:1325
texts = {optimized out}

FUNCTION = "fill_unit_type_sprite_array"

SAVE_SPRS = 0x7fffffffa510
OWNER = 0x0
DO_DRAW_UNIT = {optimized out}
SOLID_BG = {optimized out}
FUNCTION = "fill_unit_sprite_array"

SAVE_SPRS = 0x7fffffffa510
OWNER = 0x0
DO_DRAW_UNIT = {optimized out}
SOLID_BG = {optimized out}
FUNCTION = "fill_unit_sprite_array"
I can't reproduce with Qt "load ancients then civ2civ3_earth". But it smells to me like it might be a random failure after changing tilesets. Both those rulesets want custom tilesets, so probably the tileset changed. Perhaps building with ASan or running under valgrind or something would catch it.

Quick look:

Both backtraces contain hud_units::update_actions().

Looking at the second backtrace (assertion failure):

Looks like get_unittypetype_sprite() returned NULL for some reason.

The referenced unit type is "unit_types+75896". Guessing at which one that is:

- On my system, a unit_type member is 1440 bytes. (gdb ptype /o unit_type)
- factor 75896 gives 2 2 2 53 179. From those I'm guessing Chippo's client had unit_type size be 1432 bytes, and this was unit_types[53]. (Assuming that putype pointer is valid, which it might not be.)
- In my copy of civ2civ3_earth, that is Explorer, which is plausible.
- ...don't think that tells me anything.

#3 - 2020-02-01 01:04 PM - Chippo Elder

Jacob Nevins wrote:

Perhaps building with ASan or running under valgrind or something would catch it.

Wilco! I've been looking for a justifiable excuse to compile with the llvm toolchain and you've given it to me. But I saw that a Debian maintainer had some llvm-freeciv complaints and that's another excuse to give clang+freeciv a whirl.

#4 - 2020-02-01 07:57 PM - Chippo Elder

I rebuild with ASAN and followed the steps that mostly hits that assert, but before the point where it would normally print the assert message, the program exited and printed out a lot of stuff. It looks unrelated to the other two BTs.

chippo@chippo-Aspire-V3-731:~$ freeciv-qt-26 -F
2: Loading tileset "amplio2".
2: Loading tileset "delta2".
2: Loading tileset "trident".
QSocketNotifier: Invalid socket 41 and type 'Read', disabling...
2: Loading tileset "amplio2".
2: Loading tileset "delta2".
2: Loading tileset "amplio2".
2: Loading tileset "amplio_earth".
lpng warning: ICCP: known incorrect sRGB profile
lpng warning: ICCP: known incorrect sRGB profile
lpng warning: ICCP: known incorrect sRGB profile
lpng warning: ICCP: known incorrect sRGB profile
722231ERROR: AddressSanitizer: heap-use-after-free on address 0x603000a16de8 thread T0
#0 0x99c5e3 in skip_hostname_prefix /home/chippo/Downloads/git_clones/freeciv/utility/fcintl.c:48:7
I can't reproduce with Qt "load ancients then civ2civ3_earth". But it smells to me like it might be a random failure after changing tileset. Both those rulesets want custom tilesets, so probably the tileset changed.

The backtrace reveals that this happened when timer went off. So timing may matter (i/s the key random factor) in reproducibility. Likely tileset is just changing and the state is not sane when the timer goes off, and we don't notice that but go on using the tileset like it was a legal thing to do.

I assume this would help.

I've got good news, and bad news, and good news.

Firstly, this patch definitely does something good. I think we should (soon) be able to close (at least) this ticket. The bad news is administrative. With this patch applied, several bugs I ticketed recently are no longer reproducible. Or maybe I'm not trying hard enough - since you say something about timing. There's going to be a lot of admin deciding which tickets to reject, which to close and which to keep open.

The other good news is that this bug was holding me back on properly ticketing some other bugs, which I can now proceed with.

Anything that is related to a tileset changing and having update_unqueue() in the backtrace can be considered duplicates of this.
- Status changed from Resolved to Closed
- Assignee set to Marko Lindqvist

**Files**

<table>
<thead>
<tr>
<th>File Name</th>
<th>Size</th>
<th>Date</th>
<th>Assignee</th>
</tr>
</thead>
<tbody>
<tr>
<td>0007-Do-not-unqueue-mapview-updates-when-tileset-is-not-f.patch</td>
<td>1.09 KB</td>
<td>2020-02-02</td>
<td>Marko Lindqvist</td>
</tr>
</tbody>
</table>