I provoked a segfault twice while mucking around with the new "direction" module added in gna patch #5415.

It doesn't seem 100% reproducible, but it seems to happen sooner or later. I haven't worked out if it's specific to the 'direction' module or a general Lua problem or what; I haven't investigated at all.

The first segfault (for which I didn't get a coredump) was in response to

```
> /lua cmd print(direction.opposite(direction.str2dir("north")))
Segmentation fault (core dumped)
```

This exact command didn't provoke a segfault the second time, but I got one eventually:

```
2: Now accepting new client connections on port 5556.

For introductory help, type 'help'.
>
> /lua cmd print(direction.opposite(direction.str2dir("north")))
tolua.Direction: 0x233f6a8
> /lua cmd print(direction.str2dir("north"))
t tolua.Direction: 0x233fd38
> /lua cmd print(direction.opposite(direction.str2dir("north")))
t tolua.Direction: 0x233f6e8
> /lua cmd print(direction.opposite(direction.str2dir("north")))
t tolua.Direction: 0x2340818
> /lua cmd print(direction.opposite(direction.str2dir("north")))
t tolua.Direction: 0x2340228
> /lua cmd print(direction.opposite(direction.str2dir("east")))
t tolua.Direction: 0x2341468
> /lua cmd print(direction.opposite(direction.str2dir("west")))
t tolua.Direction: 0x2341938
> 
> /lua cmd print(direction.opposite(direction.str2dir("south")))
t tolua.Direction: 0x2341e58
> /lua cmd print(direction.opposite(direction.str2dir("southwest")))
t tolua.Direction: 0x2342088
> /lua cmd print(direction.cw(direction.str2dir("southwest")))
Segmentation fault (core dumped)
```

No symbol table info available.

```
#0 0x0000000000000000 in ?? ()
No symbol table info available.
#1 0x00000000000465a5f in luaD_preccall (L=L@entry=0x31cebd8, func=0x233f7d0, nresults=0) at /home/jtn/src/freeciv/git26/dependencies/lua-5.3/src/ldo.c:434
    n = <optimised out>
    f = 0x0
    ci = 0x31e1370
    __PRETTY_FUNCTION__ = "luaD_preccall"
#2 0x00000000000466d0 in luaD_call (L=L@entry=0x31cebd8, func=<optimised out>, nResults=nResults@entry=0) at /home/jtn/src/freeciv/git26/dependencies/lua-5.3/src/ldo.c:498
No locals.
#3 0x00000000000486e1e in luaD_callnoyield (L=L@entry=0x31cebd8, func=<optimised out>, nResults=nResults@entry=0) at /home/jtn/src/freeciv/git26/dependencies/lua-5.3/src/ldo.c:509
No locals.
```
I've provoked trouble on the S2_5 branch (commit:6fda16cd21) as well.

I still have investigated it adequately, but some observations:

- The number (pointer?) that's printed is different every time, but ought to represent the same direction?
- "Direction" is the only one of the Lua classes which is backed by an enum; all the others are structs.
line_internal = 0x221ce80 /*/lua cmd print(str2direction("northeast"))*/
line = 0x2215790 /*/lua cmd print(str2direction("northeast"))*/

No symbol table info available.

i = <optimised out>
s = <optimised out>
max_desc = <optimised out>
readfs = {fds_bits = {1, 0 <repeats 15 times>}}
writefs = {fds_bits = {0 <repeats 16 times>}}
exceptfs = {fds_bits = {0 <repeats 16 times>}}
tv = {tv_sec = 0, tv_usec = 940067}
__FUNCTION__ = "server_sniff_all_input"

i = <optimised out>
s = <optimised out>
max_desc = <optimised out>
readfs = {fds_bits = {1, 0 <repeats 15 times>}}
writefs = {fds_bits = {0 <repeats 16 times>}}
exceptfs = {fds_bits = {0 <repeats 16 times>}}
tv = {tv_sec = 0, tv_usec = 940067}
__FUNCTION__ = "server_sniff_all_input"

Setting target as 2.5.9 to indicate it's not a 2.5.8 blocker, but if there's a simple low-risk fix it can go in 2.5.8, of course.

Had another quick look. Still crashy. No real progress, but things I found:

- tolua is indeed malloc'ing a fresh thing every time an API returns a Direction:

  Direction tolua_ret = api_utilities_str2dir(L,str);
  void* tolua_obj = tolua_copy(tolua_S,(void*)&tolua_ret,sizeof(Direction)); /* calls malloc() */

- Equality is shallow:

  /lua cmd print(direction.str2dir("north") == direction.str2dir("north"))

false

(not very surprising, but not very helpful either)

Had another quick look. Still crashy. No real progress, but things I found:

- tolua is indeed malloc'ing a fresh thing every time an API returns a Direction:

  Direction tolua_ret = api_utilities_str2dir(L,str);
  void* tolua_obj = tolua_copy(tolua_S,(void*)&tolua_ret,sizeof(Direction)); /* calls malloc() */

- Equality is shallow:

  /lua cmd print(direction.str2dir("north") == direction.str2dir("north"))

false

(not very surprising, but not very helpful either)
- Category set to Scripting API
- Status changed from New to Closed
- Assignee set to Marko Lindqvist
- Sprint/Milestone set to 2.6.3

Fix is in Bug #880869.