Ruby master - Bug #16654

Segfault in rb_str_hash

02/25/2020 07:32 PM - nateberkopec (Nate Berkopec)

Status:	Open		
Priority:	Normal		
Assignee:			
Target version:			
ruby -v:	ruby 2.6.5p114 (2019-10-01 revision 67812) [x86_64-darwin18]	**Backport:**	2.5: UNKNOWN, 2.6: UNKNOWN, 2.7: UNKNOWN

Description

Seen in ruby 2.6.5.

```
-- C level backtrace information -------------------------------------------
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(rb_vm_bugreport+0x82) [0x10632d9c2]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(rb_bug_context+0x1d8) [0x1061940a8]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(sigsegv+0x51) [0x10629be41]
/usr/lib/system/libsystem_platform.dylib(_sigtramp+0x1d) [0x7fff67bd442d]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(ruby_sip_hash13+0xf0) [0x106257bf0]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(rb_str_hash+0x6c) [0x1062b035c]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(st_general_delete+0x27) [0x1062a5287]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(rb_str_free+0x61) [0x1062b3401]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(obj_free+0x270) [0x1061bbf30]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(gc_sweep_step+0x24c) [0x1061bb66c]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(gc_sweep+0xdb) [0x1061bd37b]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(gc_start+0xa48) [0x1061bafc8]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(gc_start_internal+0x282) [0x1061b7a32]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(vm_call_cfunc+0x15b) [0x1063210fb]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(vm_exec_core+0x32aa) [0x10630b19a]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(rb_vm_exec+0xa0c) [0x10631c04c]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(ruby_exec_internal+0xd8) [0x10619eeb8]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(ruby_run_node+0x49) [0x10619ed49]
/Users/nateberkopec/.rubies/ruby-2.6.5/bin/ruby(main+0x5d) [0x10611775d]
```

Another dump posted here:

https://github.com/SamSaffron/memory_profiler/issues/83

For reproduction, I was wrapping MemoryProfiler around a Rails application startup, but it may be possible to repro this in a much simpler way.

```
require "memory_profiler"
report = MemoryProfiler.report do
  require ::File.expand_path('../config/environment', __FILE__)
end

report.pretty_print
```

History

#1 - 02/25/2020 07:34 PM - nateberkopec (Nate Berkopec)

This reproduces on 2.7.0 as well (though the thing calling rb_str_hash was different, this time objspace_each_objects)

```
-- C level backtrace information -------------------------------------------
/Users/nateberkopec/.rubies/ruby-2.7.0/bin/ruby(rb_vm_bugreport+0x96) [0x10089ef76]
/Users/nateberkopec/.rubies/ruby-2.7.0/bin/ruby(rb_bug_for_fatal_signal+0x1d1) [0x1006e5131]
/Users/nateberkopec/.rubies/ruby-2.7.0/bin/ruby(sigsegv+0x5b) [0x100805b4b]
/usr/lib/system/libsystem_platform.dylib(_sigtramp+0x1d) [0x7fff67bd442d]
/Users/nateberkopec/.rubies/ruby-2.7.0/bin/ruby(ruby_sip_hash13+0xf0) [0x1007c0960]
/Users/nateberkopec/.rubies/ruby-2.7.0/bin/ruby(rb_str_hash+0x6c) [0x100819fec]
/Users/nateberkopec/.rubies/ruby-2.7.0/bin/ruby(rb_any_hash+0xfb) [0x10071ddcb]
/Users/nateberkopec/.rubies/ruby-2.7.0/bin/ruby(rb_st_lookup+0x26) [0x10080dc06]
/Users/nateberkopec/.rubies/ruby-2.7.0/bin/ruby(rb_hash_aref+0x9f) [0x1007171df]
```

```
/Users/nateberkopec/.rubies/ruby-2.7.0/bin/ruby(vm_exec_core+0x783a) [0x10087bbfa]
/Users/nateberkopec/.rubies/ruby-2.7.0/bin/ruby(rb_vm_exec+0xa13) [0x10088c103]
/Users/nateberkopec/.rubies/ruby-2.7.0/bin/ruby(rb_yield+0xa7) [0x100885307]
/Users/nateberkopec/.rubies/ruby-2.7.0/bin/ruby(os_obj_of_i+0xb7) [0x100714af7]
/Users/nateberkopec/.rubies/ruby-2.7.0/bin/ruby(objspace_each_objects_protected+0xa1) [0x1007106e1]
/Users/nateberkopec/.rubies/ruby-2.7.0/bin/ruby(rb_ensure+0xf0) [0x1006f1480]
/Users/nateberkopec/.rubies/ruby-2.7.0/bin/ruby(rb_objspace_each_objects+0x117) [0x1007004d7]
```

#2 - 02/25/2020 08:03 PM - nateberkopec (Nate Berkopec)

Does not reproduce on 2.5.5, and reproduces on 2.6.0

Unfortunately, doesn't reproduce with a simple script, e.g.:

```
require "memory_profiler"
report = MemoryProfiler.report do
  1_000_000.times { "allocate a string" }
end

report.pretty_print
```

#3 - 02/26/2020 05:29 AM - nobu (Nobuyoshi Nakada)

- Status changed from Open to Feedback

- Description updated

I couldn't reproduce it with a simple file.
Could you try with the master, and could your "config/environment.rb" be shared?

#4 - 02/26/2020 07:11 AM - sam.saffron (Sam Saffron)

Nobu, This looks like Mac to me maybe llvm related?

#5 - 02/26/2020 07:40 PM - nateberkopec (Nate Berkopec)

The application I'm reproducing it on is here: https://github.com/codetriage/codetriage

I think you should be able to "bundle install" and then use my script above and it would reproduce.

#6 - 07/31/2020 04:03 AM - mame (Yusuke Endoh)

- Status changed from Feedback to Open

Files

diagnostic_report.crash	44.1 KB	02/25/2020	nateberkopec (Nate Berkopec)
rubycrashdump.txt	3.47 KB	02/25/2020	nateberkopec (Nate Berkopec)