Ruby master - Bug #11668
SEGV instead of SystemStackError when using the TracePoint API
11/09/2015 04:38 AM - deivid (David Rodríguez)

Status: Closed
Priority: Normal
Assignee:
Target version: ruby -v:
ruby 2.3.0dev (2015-11-09 trunk 52499)
[x86_64-linux]
Backport: 2.0.0: UNKNOWN, 2.1: UNKNOWN, 2.2: UNKNOWN

Description

fixnum_monkey.rb

global Fixnum

define_method(:to_s) do |*args|
  to_default_s(*args)
end

alias_method :to_default_s, :to_s
end

If I use this code normally it gives me SystemStackError as expected

$ ruby -e "load('fixnum_monkey.rb'); 5.to_s"
fixnum_monkey.rb:2:in `block in <class:Fixnum>': stack level too deep (SystemStackError)
  from fixnum_monkey.rb:3:in `block in <class:Fixnum>'
  from fixnum_monkey.rb:3:in `block in <class:Fixnum>'
  from fixnum_monkey.rb:3:in `block in <class:Fixnum>'
  from fixnum_monkey.rb:3:in `block in <class:Fixnum>'
  from fixnum_monkey.rb:3:in `block in <class:Fixnum>'
  from fixnum_monkey.rb:3:in `block in <class:Fixnum>'
  from fixnum_monkey.rb:3:in `block in <class:Fixnum>'
  ... 5804 levels...
  from fixnum_monkey.rb:3:in `block in <class:Fixnum>'
  from fixnum_monkey.rb:3:in `block in <class:Fixnum>'
  from fixnum_monkey.rb:3:in `block in <class:Fixnum>'
  from fixnum_monkey.rb:3:in `block in <class:Fixnum>'
  from -e:1:in `<main>'

But if I enable the return event of the TracePoint API before using it, I get a segfault

$ ruby -e "load('fixnum_monkey.rb'); TracePoint.trace(:return) { |_tp| }; 5.to_s"

Violación de segmento

History

#1 - 10/21/2019 05:17 PM - jeremyevans0 (Jeremy Evans)
- Status changed from Open to Closed

It looks like this problem was fixed. While I can get this code to segfault on Ruby <2.5, I can no longer get a segfault for this code starting with Ruby 2.5:

class Fixnum
  define_method(:to_s) do |*args|
    to_default_s(*args)
  end

  alias_method :to_default_s, :to_s
end

TracePoint.trace(:return) { |_tp| }
1.to_s