

Ruby - Bug #9937

Segfault in the TracePoint API

06/12/2014 04:33 PM - deivid (David Rodríguez)

Status:	Closed	
Priority:	Normal	
Assignee:	ko1 (Koichi Sasada)	
Target version:		
ruby -v:	ruby 2.2.0dev (2014-06-12 trunk 46413) [i686-linux]	
		Backport: 2.0.0: UNKNOWN, 2.1: UNKNOWN
Description		
Hi!		
It seems that a regression bug has been recently introduced in the TracePoint API (maybe after changeset r45758). byebug's test suite works against 2.0 and 2.1, but segfaults against ruby-head. See https://travis-ci.org/deivid-rodriguez/byebug		
Segfault occurs in https://github.com/deivid-rodriguez/byebug/blob/master/ext/byebug/byebug.c#L252 , I'm guessing the method ID provided by the TracePoint API is nil, so the SYM2ID call fails.		
I'm sorry I'm not currently able to properly investigate this.		

History

#1 - 06/15/2014 07:03 PM - deivid (David Rodríguez)

Hi [@ko1 \(Koichi Sasada\)](#), I see that you made some changes in the TracePoint API, but the build is still segfaulting at the same place. Just to confirm this is still an issue.

Thanks!

#2 - 06/25/2014 01:29 PM - deivid (David Rodríguez)

Bump!

Similar update: after latest bug fixes in the API ([#9957](#), [#9959](#), [#9961](#), [#9964](#)), the build still segfaults in the same place.

By the way, thanks for all the recent work and bug fixes in the TracePoint API!!!

#3 - 07/31/2014 02:38 AM - nagachika (Tomoyuki Chikanaga)

- Status changed from Open to Assigned

Thank you for your report.

ko1 san, could you handle this issue?

#4 - 07/31/2014 01:37 PM - deivid (David Rodríguez)

Hi, I'll be fixing this inside byebug. I don't think the assumption that TracePoint API events will always provide a method_id is correct, so I'll be checking for nil values.

I guess this can be closed.

#5 - 11/27/2014 11:42 AM - ko1 (Koichi Sasada)

sorry for late.

Can you make small test for it? easy to keep away from this issue, but it should be fixed.

#6 - 11/30/2014 02:01 AM - deivid (David Rodríguez)

Hi Koichi, I've revisited this. This is not a bug in the TracePoint API. it was a bug in Byebug. I was using the same callback for the RUBY_EVENT_B_CALL, RUBY_EVENT_CLASS and RUBY_EVENT_CALL, but only the RUBY_EVENT_CALL provides a method_id, so the other events were failing when trying to grab the method_id.

The issue can be closed.

#7 - 12/01/2014 04:53 AM - ko1 (Koichi Sasada)

- Status changed from Assigned to Closed

Ah, I see.

You can't get method_id for top level.

```
TracePoint.new(:class, :b_call){|tp|
  p [tp.event, tp.method_id]
}.enable
1.times{}
class C; end
def foo; 1.times{}; end; foo

#=>
ruby 2.2.0dev (2014-11-27 trunk 48608) [i386-mswin32_110]
[:b_call, nil]
[:class, nil]
[:b_call, :foo]
```

#8 - 12/01/2014 01:12 PM - deivid (David Rodríguez)

Exactly. :)