

Ruby - Bug #4881

super() in define_method which returns proc causes SEGV

06/14/2011 05:52 AM - enebo (Thomas Enebo)

Status:	Closed	Backport:
Priority:	Normal	
Assignee:		
Target version:		
ruby -v:	ruby 1.9.2p180 (2011-02-18 revision 30909) [x86_64-darwin10.7.0]	
Description		
In working through some corner cases, we ran across (https://gist.github.com/1023643):		
<pre>class Base def foo puts "Base foo" end def bar puts "Base bar" end end class Derived < Base def bar self.class.send(:define_method, :foo) do proc { super() } end end end d = Derived.new code = d.bar d.foo.call</pre>		
And this ends up generating a SEGV:		
<pre>ruby super2.rb super2.rb:15: [BUG] Segmentation fault ruby 1.9.2p180 (2011-02-18 revision 30909) [x86_64-darwin10.7.0]</pre>		
-- control frame -----		
c:0006 p:0008 s:0016 b:0015 l:000a18 d:000014 BLOCK super2.rb:15		
c:0005 p:---- s:0013 b:0013 l:000012 d:000012 FINISH		
c:0004 p:---- s:0011 b:0011 l:000010 d:000010 CFUNC :call		
c:0003 p:0075 s:0008 b:0008 l:0014f8 d:0003c8 EVAL super2.rb:22		
c:0002 p:---- s:0004 b:0004 l:000003 d:000003 FINISH		
c:0001 p:0000 s:0002 b:0002 l:0014f8 d:0014f8 TOP		
 -- Ruby level backtrace information -----		
super2.rb:22:in <main>' super2.rb:22:in call'		
super2.rb:15:in `block (2 levels) in bar'		
 -- C level backtrace information -----		
 [NOTE]		
You may have encountered a bug in the Ruby interpreter or extension libraries.		
Bug reports are welcome.		
For details: http://www.ruby-lang.org/bugreport.html		

This should print out that it is calling foo in the base class.

Associated revisions

Revision 9c931b85 - 06/25/2011 03:05 PM - nagachika (Tomoyuki Chikanaga)

- vm_insnhelper.c (vm_search_superclass): avoid control frame stack overrun. currently super() in Proc created in a method defined by Module#define_method raise NoMethodError. [Bug #4881]
- test/ruby/test_method.rb t_super_in_proc_from_define_method): add test for it.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@32227 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision c88e3861 - 08/07/2011 10:03 AM - yugui (Yuki Sonoda)

merges 32227 from trunk into ruby_1_9_2.

- vm_insnhelper.c (vm_search_superclass): avoid control frame stack overrun. currently super() in Proc created in a method defined by Module#define_method raise NoMethodError. [Bug #4881]
- test/ruby/test_method.rb t_super_in_proc_from_define_method): add test for it.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_1_9_2@32881 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 02d5868962417298cf041ce723d406e16a635693 - 07/06/2015 06:44 PM - ko1 (Koichi Sasada)

- vm_core.h: remove rb_iseq_t::klass to reduce dynamic data.
- internal.h, iseq.c (rb_iseq_klass): remove it because rb_iseq_t::klass is removed.
- vm_insnhelper.c (vm_super_outside): do not see cfp->iseq, but check callable method entry on a frame.
This fix simplify the logic to search super class.
- test/ruby/test_method.rb: support super() from Proc.
Now, [Bug #4881] and [Bug #3136] was solved.
- proc.c (rb_mod_define_method): catch up this change.
- vm.c (vm_define_method): ditto.
- vm_backtrace.c (rb_profile_frames): now, each `frame' objects are rb_callable_method_entry_t data or iseq VALUEs.
This fix introduce minor compatibility issue that rb_profile_frame_label() always returns rb_profile_frame_base_label().
- test/-ext-/debug/test_profile_frames.rb: catch up this change.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@51166 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 02d58689 - 07/06/2015 06:44 PM - ko1 (Koichi Sasada)

- vm_core.h: remove rb_iseq_t::klass to reduce dynamic data.
- internal.h, iseq.c (rb_iseq_klass): remove it because rb_iseq_t::klass is removed.
- vm_insnhelper.c (vm_super_outside): do not see cfp->iseq, but check callable method entry on a frame.
This fix simplify the logic to search super class.
- test/ruby/test_method.rb: support super() from Proc.
Now, [Bug #4881] and [Bug #3136] was solved.
- proc.c (rb_mod_define_method): catch up this change.
- vm.c (vm_define_method): ditto.
- vm_backtrace.c (rb_profile_frames): now, each `frame' objects are rb_callable_method_entry_t data or iseq VALUEs.
This fix introduce minor compatibility issue that rb_profile_frame_label() always returns rb_profile_frame_base_label().
- test/-ext-/debug/test_profile_frames.rb: catch up this change.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@51166 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 06/25/2011 10:00 PM - nagachika (Tomoyuki Chikanaga)

- File `raise_super_outside_of_method.patch` added
- Category set to `core`

Hi,

Thank you for your reporting.
This issue seems similar to [#2420](#) and [#3136](#) .
According to discussion in [#3136](#), fundamental fix needs tough work.

I've wrote quick fix patch for only avoid SEGV but raise an `NoMethodError` exception.

#2 - 06/26/2011 12:05 AM - nagachika (Tomoyuki Chikanaga)

- Status changed from `Open` to `Closed`
- % Done changed from `0` to `100`

This issue was solved with changeset `r32227`.
Thomas, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

-
- `vm_insnhelper.c (vm_search_superclass)`: avoid control frame stack overrun. currently `super()` in `Proc` created in a method defined by `Module#define_method` raise `NoMethodError`. [Bug [#4881](#)]
 - `test/ruby/test_method.rb t_super_in_proc_from_define_method`: add test for it.

Files

<code>raise_super_outside_of_method.patch</code>	587 Bytes	06/25/2011	nagachika (Tomoyuki Chikanaga)
--	-----------	------------	--------------------------------