Backport193 - Backport #7123

Segmentation fault in ruby 1.9.3-p194

10/09/2012 10:16 AM - mscottford (M. Scott Ford)

Status:	Closed	
Priority:	Normal	
Assignee:	authorNari (Narihiro Nakamura)	

Description

Example source for this issue is posted at https://github.com/mscottford/segfault-test, with reproduction instructions.

I'm encountering a segmentation fault in ruby 1.9.3-p194 on a project using Rails 3.2.8. The issue is only happening on Mac OS X. Members of my team that are running Linux do not have the same issue. The issue does not occur consistently; it sometimes takes several (20+) runs for the crash to happen.

test:

```
require 'spec_helper'

describe Widget do
  it "removes widget on rejection" do
    widget = Widget.create!

    expect do
      widget.reject!
    end.to change { described_class.count }.by(-1)

    GC.start
  end
end
```

model:

```
class Widget < ActiveRecord::Base
  attr_accessor :check_rejection_reason

  state_machine :initial => :requested do
    # I suspect that the issue is related to the issue being accessed in this closure after it has
 been deleted
    around_transition :requested => :none do |gm, transition, blk|
      gm.check_rejection_reason = true
      blk.call
      gm.check_rejection_reason = false
    end

    # This closure deletes the instance, but it is still being accessed by the `around_transition`
 above.
    after_transition any => :none do |gm, transition|
      gm.destroy
    end

    on :reject do
      transition :requested => :none
    end
  end
end
```

Associated revisions

Revision ae2df330 - 11/13/2012 10:02 AM - usa (Usaku NAKAMURA)

merged revision(s) 37075,37076,37082,37083,37088: [Backport #7123]

- gc.c: Use the non-recursive marking instead of recursion. The recursion marking of CRuby needs checking stack overflow and the

fail-safe system, but these systems not good at partial points,
for example, marking deep tree structures. [ruby-dev:46184]
[Feature #7095]

- configure.in (GC_MARK_STACKFRAME_WORD): removed. It's used by checking stack overflow of marking.
- win32/Makefile.sub (GC_MARK_STACKFRAME_WORD): ditto.
- gc.c (free_stack_chunks): it is used only when per-VM object space is enabled.
- gc.c (rb_objspace_call_finalizer): mark self-referencing finalizers before run finalizers, to fix SEGV from btest on 32bit.
- gc.c (gc_mark_stacked_objects): extract from gc_marks().
- gc.c (rb_objspace_call_finalizer): call gc_mark_stacked_objects at suitable point.
- gc.c (init_heap): call init_mark_stack before to allocate altstack. This change avoid the stack overflow at the signal handler on 32bit, but I don't understand reason... [Feature #7095]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_1_9_3@37648 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 37648 - 11/13/2012 10:02 AM - usa (Usaku NAKAMURA)

merged revision(s) 37075,37076,37082,37083,37088: [Backport #7123]

- gc.c: Use the non-recursive marking instead of recursion. The recursion marking of CRuby needs checking stack overflow and the fail-safe system, but these systems not good at partial points, for example, marking deep tree structures. [ruby-dev:46184] [Feature #7095]
- configure.in (GC_MARK_STACKFRAME_WORD): removed. It's used by checking stack overflow of marking.
- win32/Makefile.sub (GC_MARK_STACKFRAME_WORD): ditto.
- gc.c (free_stack_chunks): it is used only when per-VM object space is enabled.
- gc.c (rb_objspace_call_finalizer): mark self-referencing finalizers before run finalizers, to fix SEGV from btest on 32bit.
- gc.c (gc_mark_stacked_objects): extract from gc_marks().
- gc.c (rb_objspace_call_finalizer): call gc_mark_stacked_objects at suitable point.
- gc.c (init_heap): call init_mark_stack before to allocate altstack. This change avoid the stack overflow at the signal handler on 32bit, but I don't understand reason... [Feature #7095]

History

#1 - 10/09/2012 10:38 PM - mscottford (M. Scott Ford)

I got a report that the link is broken because it's including a comma. Here's the correct link: https://github.com/mscottford/segfault-test

#2 - 10/17/2012 07:21 PM - rsluiters (Ralph Sluiters)

- File error_log.txt added

I also get a segmentation fault in 1.9.3 (p0 and p268), especially when doing UI operations in our complex Rails App. The bug vanishes whenever I switch off the garbage collector. Is this the case for you as well, then it might be the same bug...

#3 - 10/25/2012 11:31 AM - seangeo (Sean Geoghegan)

- File Bug 7123 - seangeo.crash added

I've also experience the same issue.

We also have a model using state_machine with an around transition and during testing it will crash with a segmentation fault in the GC stack about 10% of the time. If we remove the around_transition there are no longer any crashes when running tests.

However, I'm the only one in my team to experience it. I'm using OSX 10.7.3 and other on my team are using a mix of 10.7.4, 10.7.5 and 10.8.x. This has happened with Ruby 1.9.3 p125 and p194. I've attached the crash log for the error.

#4 - 11/06/2012 09:36 PM - mame (Yusuke Endoh)

- Tracker changed from Bug to Backport

- Project changed from Ruby trunk to Backport193

- Status changed from Open to Assigned

- Assignee set to usa (Usaku NAKAMURA)

Looks stack overflow in GC.
I believe that this was fundamentally fixed by removing recursive calls from GC marking phase.
Please let me know if it occurs on trunk or 2.0.0-preview1.

I'm moving this ticket to 1.9.3 tracker.

Maybe related to #6577, #7141, and #7095.

--
Yusuke Endoh mame@tsg.ne.jp

#5 - 11/06/2012 09:38 PM - usa (Usaku NAKAMURA)

- Assignee changed from usa (Usaku NAKAMURA) to authorNari (Narihiro Nakamura)

nari3, can you make a patch for 1.9.3?

#6 - 11/10/2012 02:56 PM - authorNari (Narihiro Nakamura)

Ummm... I can create a patch, but is it needed?
The purpose of the Non-recursive marking is not only a bug fix.

#7 - 11/12/2012 04:07 PM - usa (Usaku NAKAMURA)

If the patch does not changes the behavior of ruby, it's OK.
"Not changes the behavior" means that there is no ABI changes, and it passes test, test-all and rubyspec.

#8 - 11/13/2012 06:13 PM - authorNari (Narihiro Nakamura)

- File backport_r37088_r37083_r37082_r37076_r37075_to_193.patch added

usa (Usaku NAKAMURA) wrote:

> If the patch does not changes the behavior of ruby, it's OK.
> "Not changes the behavior" means that there is no ABI changes, and it passes test, test-all and rubyspec.

I see. I've created the backport patch for r37088,r37083,r37082,r37076,r37075.
Could you check it?

Thank you!

#9 - 11/13/2012 07:02 PM - usa (Usaku NAKAMURA)

- Status changed from Assigned to Closed

- % Done changed from 0 to 100

This issue was solved with changeset r37648.
M. Scott, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

---

merged revision(s) 37075,37076,37082,37083,37088: [Backport #7123]

- gc.c: Use the non-recursive marking instead of recursion. The

recursion marking of CRuby needs checking stack overflow and the fail-safe system, but these systems not good at partial points, for example, marking deep tree structures. [ruby-dev:46184] [Feature #7095]

- configure.in (GC_MARK_STACKFRAME_WORD): removed. It's used by checking stack overflow of marking.
- win32/Makefile.sub (GC_MARK_STACKFRAME_WORD): ditto.
- gc.c (free_stack_chunks): it is used only when per-VM object space is enabled.
- gc.c (rb_objspace_call_finalizer): mark self-referencing finalizers before run finalizers, to fix SEGV from btest on 32bit.
- gc.c (gc_mark_stacked_objects): extract from gc_marks().
- gc.c (rb_objspace_call_finalizer): call gc_mark_stacked_objects at suitable point.
- gc.c (init_heap): call init_mark_stack before to allocate altstack. This change avoid the stack overflow at the signal handler on 32bit, but I don't understand reason... [Feature #7095]

#10 - 11/13/2012 10:20 PM - usa (Usaku NAKAMURA)

nari3, I committed your patch and RubyCI says it's OK.
Thank you for your kindly help!

#11 - 11/23/2012 07:01 PM - saurabhnanda (Saurabh Nanda)

usa (Usaku NAKAMURA) wrote:

> nari3, I committed your patch and RubyCI says it's OK.
> Thank you for your kindly help!

I'm facing the same issue on Mac OSX with ruby-1.9.3p327

Tests are segfaulting about randomly whenever state_machine has an around_transition definition.

How do I apply a fix to my version of ruby?

#12 - 11/23/2012 09:04 PM - saurabhnanda (Saurabh Nanda)

> How do I apply a fix to my version of ruby?

I compiled and test the 1.9.3-head which apparently has this patch. Thanks for fixing this guys!

#13 - 07/11/2013 02:29 AM - yopp (Alex Yopp)

Hi.

Seems like this issue is still there.

I can confirm that test case provided by M. Scott Ford is failing on "ruby 1.9.3p448 (2013-06-27 revision 41675) [x86_64-darwin13.0.0]". We will check on other configurations as well.

Update: This issue is not reproducible on "ruby 2.0.0p247 (2013-06-27 revision 41674) [x86_64-darwin13.0.0]".

#14 - 07/11/2013 07:37 AM - yopp (Alex Yopp)

It's also reproducible on release version of OSX:

ruby 1.9.3p392 (2013-02-22 revision 39386) [x86_64-darwin12.3.0]
12.4.0 Darwin Kernel Version 12.4.0: Wed May 1 17:57:12 PDT 2013; root:xnu-2050.24.15~1/RELEASE_X86_64 x86_64

I suggest to reopen this issue to validate applied patches. According to current 1.9.3 ChangeLog, they should be included in p392 and higher.

Thank you.

Files

ruby_2012-10-08-204054_cloudraker.crash	39.1 KB	10/09/2012	mscottford (M. Scott Ford)
error_log.txt	211 KB	10/17/2012	rsluiters (Ralph Sluiters)
Bug 7123 - seangeo.crash	64.7 KB	10/25/2012	seangeo (Sean Geoghegan)
backport_r37088_r37083_r37082_r37076_r37075_to_193.patch	21.7 KB	11/13/2012	authorNari (Narihiro Nakamura)