Ruby master - Bug #4492

Segfault on successive stack overflows

03/10/2011 10:39 PM - catwell (Pierre Chapuis)

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td>ko1 (Koichi Sasada)</td>
</tr>
<tr>
<td>Target version:</td>
<td>2.6</td>
</tr>
<tr>
<td>ruby -v:</td>
<td>ruby 1.9.2p136 (2010-12-25 revision 30365) [x86_64-darwin10.6.0]</td>
</tr>
<tr>
<td>Backport:</td>
<td>2.2: UNKNOWN, 2.3: UNKNOWN, 2.4: UNKNOWN</td>
</tr>
</tbody>
</table>

Description

On a Mac, this snippet crashes Ruby:

```ruby
def f(*x); end
begin
  f(*(0..120000))
rescue
  nil
end
f(*(0..120000))
```

Here is the result when run with ruby:

```
segfaulter.rb:7: [BUG] Segmentation fault
ruby 1.9.2p136 (2010-12-25 revision 30365) [x86_64-darwin10.6.0]
-- control frame ----------
c:0003 p:0048 s:120084 b:0006 l:001af8 d:000c08 EVAL segfaulter.rb:7
```

```
-- Ruby level backtrace information ----------------------------------------
segfaulter.rb:7:in `<main>'
```

```
-- 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
```

Abort trap

Here is the result in irb:

```
ruby-1.9.2-p136 :001 > def f(*x); end
 => nil
ruby-1.9.2-p136 :002 > begin
ruby-1.9.2-p136 :003 >     f(*(0..120000))
ruby-1.9.2-p136 :004> rescue
ruby-1.9.2-p136 :005>    nil
ruby-1.9.2-p136 :006> end
 => nil
ruby-1.9.2-p136 :007 > f(*(0..120000))
(irb):7: [BUG] Bus Error
ruby 1.9.2p136 (2010-12-25 revision 30365) [x86_64-darwin10.6.0]
-- control frame ----------
```

```
c:0022 p:---- s:0080 b:0080 l:000079 d:000079 FINISH
```
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

Abort trap

I cannot reproduce this on an EC2 instance running Linux, where 'ruby -v' gives:

ruby 1.9.2p136 (2010-12-25 revision 30365) [x86_64-linux]

---

History

#1 - 03/10/2011 10:44 PM - catwell (Pierre Chapuis)

Oops, apparently Redmine didn't like my markup. Same report with an alternative syntax follows...

On a Mac, this snippet crashes Ruby:

```ruby
def f(x): end
begin
  f((0..120000))
rescue
  nil
end
f*(0..120000))
```

Here is the result when run with ruby:

```
03/17/2022
```

---

03/17/2022
Here is the result in irb:

```ruby
ruby-1.9.2-p136 :001 > def f(x); end
=> nil
ruby-1.9.2-p136 :002 > begin
ruby-1.9.2-p136 :003 >       f((0..120000))
ruby-1.9.2-p136 :004?>   rescue
ruby-1.9.2-p136 :005?>     nil
ruby-1.9.2-p136 :006?>   end
=> nil
ruby-1.9.2-p136 :007 > f(*(0..120000))
```

Here is the result in irb:

```ruby
ruby-1.9.2-p136 :001 > def f(x); end
=> nil
ruby-1.9.2-p136 :002 > begin
ruby-1.9.2-p136 :003 >       f((0..120000))
ruby-1.9.2-p136 :004?>   rescue
ruby-1.9.2-p136 :005?>     nil
ruby-1.9.2-p136 :006?>   end
=> nil
ruby-1.9.2-p136 :007 > f(*(0..120000))
```
[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

Abort trap
I cannot reproduce this on a EC2 instance running Linux, where 'ruby -v' gives:

ruby 1.9.2p136 (2010-12-25 revision 30365) [x86_64-linux]
=end

#2 - 06/26/2011 06:33 PM - naruse (Yui NARUSE)
- Status changed from Open to Assigned
- Assignee set to ko1 (Koichi Sasada)
- Target version set to 2.0.0

#3 - 11/26/2012 09:16 AM - ko1 (Koichi Sasada)
- Status changed from Assigned to Feedback
- Target version changed from 2.0.0 to 2.6

#4 - 12/13/2012 10:10 PM - nobu (Nobuyoshi Nakada)
It was fixed.