

## Backport187 - Bug #3143

### [Ruby 1.8.7] Segmentation fault / stack level too deep

04/14/2010 12:01 AM - danielribeiro (Daniel Ribeiro)

<b>Status:</b>	Closed
<b>Priority:</b>	Normal
<b>Assignee:</b>	
<b>Target version:</b>	
<b>ruby -v:</b>	ruby 1.8.7 (2008-08-11 patchlevel 72) [i486-linux]

#### Description

=begin

While playing around with the possibility of a simpler AOP library for ruby than Aquarium (and without using ParseTree), I've found a very strange segmentation fault. The attached code file shows it happening. As it is very exploratory code, it is filled with some very unusual meta-programming that, when cleaned up with proper modularization, prevent the bug from happening.

What is weird is that commenting a few lines of the code change the segmentation fault into the following error:

```
meta_define_inline_blockeasy_seg_fault.rb:46:in `new': stack level too deep (SystemStackError)
```

The commented lines are marked in the file, but I repeat them for completeness sake:

```
## Commenting the following two lines the seg fault turn into `new': stack level too deep
```

```
regex = /#<Module:0x(\w+)/
```

```
uniquename = regex.match(to_s)[1]
```

```
##
```

Not a big issue, as the real library will not open Class and redefine new, but I found interesting. And worried, since seg faults can sometimes be exploited into bigger issues.

=end

#### History

##### #1 - 04/14/2010 08:01 AM - danielribeiro (Daniel Ribeiro)

=begin

Just tried on a newer patch of ruby 1.8.7:

```
$ ruby -v
```

```
ruby 1.8.7 (2009-04-08 patchlevel 160) [i686-linux]
```

The difference is that commenting the highlighted lines does not change the error from "seg fault" to "stack level too deep" anymore.

=end

##### #2 - 08/27/2019 04:37 PM - jeremyevans0 (Jeremy Evans)

- Status changed from Open to Closed

- Description updated

#### Files

easy_seg_fault.rb	1.28 KB	04/14/2010	danielribeiro (Daniel Ribeiro)
-------------------	---------	------------	--------------------------------