

Ruby trunk - Bug #15638

Another race conditions in MonitorMixin when interrupted?

03/05/2019 10:46 AM - adrienjarthon (Adrien Jarthon)

Status: Open	
Priority: Normal	
Assignee:	
Target version:	
ruby -v: ruby 2.6.1p33 (2019-01-30 revision 66950) [x86_64-linux]	Backport: 2.4: UNKNOWN, 2.5: UNKNOWN, 2.6: UNKNOWN

Description

Hi, I believe I've got a race condition issue in MonitorMixin when interrupted (by TimeoutError in my example). It looks very similar to <https://bugs.ruby-lang.org/issues/14998#change-75226> but unfortunately it looks like this fix landed in 2.6.1 and I still have the issue in 2.6.1 ☹

I have the issue with all 2.5.x and 2.6.x releases, on 2.4 and 2.3 it seems to work fine though (I didn't try below).

I managed to make a small script to reproduce the issue easily:

```
#!/usr/bin/env ruby
require 'timeout'

class Printer
  include MonitorMixin
  def write(message)
    synchronize do
      puts message
      sleep 0.15
    end
  end
end

$printer = Printer.new

def printer_with_timeout timeout: 0.1
  Timeout::timeout(timeout) do
    $printer.write "Printing"
  end
rescue Timeout::Error
  $stderr.puts "Timeout Error"
end

# Avoid double backtrace at the end
Thread::report_on_exception = false
puts "Ruby: #{RUBY_VERSION}"
25.times do |i|
  t1 = Thread.new {
    printer_with_timeout timeout: 0.1
  }
  t2 = Thread.new {
    sleep 0.05
    printer_with_timeout timeout: 0.05
    $printer.write "Worked this time (#{i})"
  }

  t1.join; t2.join
end
```

I put a loop because I couldn't find a way to reproduce it all the time, but with this code and loop I reproduce it, like 90% of the time. Here are a couple examples of failing output (it seems to fail when the second thread is interrupted while waiting for the mutex lock):

```
./test.rb
Ruby: 2.5.3
Printing
```

```
Timeout Error
Printing
Timeout Error
Worked this time (0)
Printing
Timeout Error
Timeout Error
#<Thread:0x00007fd429022e98@./test.rb:29 run> terminated with exception (report_on_exception is true):
Traceback (most recent call last):
  4: from ./test.rb:32:in `block (2 levels) in <main>'
  3: from ./test.rb:7:in `write'
  2: from /home/adrien/.rbenv/versions/2.5.3/lib/ruby/2.5.0/monitor.rb:224:in `mon_synchronize'
  1: from /home/adrien/.rbenv/versions/2.5.3/lib/ruby/2.5.0/monitor.rb:185:in `mon_enter'
/home/adrien/.rbenv/versions/2.5.3/lib/ruby/2.5.0/monitor.rb:185:in `lock': deadlock; recursive locking (ThreadError)
Traceback (most recent call last):
  4: from ./test.rb:32:in `block (2 levels) in <main>'
  3: from ./test.rb:7:in `write'
  2: from /home/adrien/.rbenv/versions/2.5.3/lib/ruby/2.5.0/monitor.rb:224:in `mon_synchronize'
  1: from /home/adrien/.rbenv/versions/2.5.3/lib/ruby/2.5.0/monitor.rb:185:in `mon_enter'
/home/adrien/.rbenv/versions/2.5.3/lib/ruby/2.5.0/monitor.rb:185:in `lock': deadlock; recursive locking (ThreadError)
```

Reproduces on: 2.5.0, 2.5.1, 2.5.3, 2.6.0, 2.6.1

I do not reproduce on: 2.4.5, 2.4.4, 2.4.1, 2.3.1, 2.3.0