Ruby trunk - Bug #15082

Memory leak in net/http/response and net/http/header

09/06/2018 06:23 AM - alexis (Alexis Bernard)

Status:	Open		
Priority:	Normal		
Assignee:			
Target version:			
ruby -v:	ruby 2.5.1p57 (2018-03-29 revision 63029) [x86_64-linux]	**Backport:**	2.3: UNKNOWN, 2.4: UNKNOWN, 2.5: UNKNOWN

Description

Hello,

I'm observing a memory leak in net/http with the following script :

```
require "net/http"
require "bundler/inline"

gemfile do
  gem "memory_profiler"
end

def profile_http_get(n)
  uri = URI("http://www.ruby-lang.org/fr/")
  http = Net::HTTP.new(uri.host, uri.port)

  report = MemoryProfiler.report do
    n.times { http.request(Net::HTTP::Get.new(uri.path)) }
  end
  report.pretty_print
end
```

Here is the MemoryProfiler report when n is 10 :

```
retained memory by gem
-----------------------------------
      4591  net

retained memory by file
-----------------------------------
      4005  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb
       586  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/header.rb

retained memory by location
-----------------------------------
      2048  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb:55
      1917  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb:62
       520  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/header.rb:83
        66  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/header.rb:68
        40  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb:43

retained memory by class
-----------------------------------
      4591  String

retained objects by gem
-----------------------------------
        67  net

retained objects by file
-----------------------------------
        53  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb
```

```
        14  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/header.rb

retained objects by location
------------------------------------
        27  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb:55
        25  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb:62
        13  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/header.rb:83
         1  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/header.rb:68
         1  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb:43

retained objects by class
------------------------------------
        67  String
```

When n is 20 :

```
retained memory by gem
------------------------------------
      8229  net

retained memory by file
------------------------------------
      7217  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb
      1012  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/header.rb

retained memory by location
------------------------------------
      3654  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb:55
      3523  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb:62
       880  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/header.rb:83
        66  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/header.rb:68
        66  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/header.rb:86
        40  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb:43

retained memory by class
------------------------------------
      8229  String

retained objects by gem
------------------------------------
       119  net

retained objects by file
------------------------------------
        95  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb
        24  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/header.rb

retained objects by location
------------------------------------
        48  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb:55
        46  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb:62
        22  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/header.rb:83
         1  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/header.rb:68
         1  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/header.rb:86
         1  /home/alexis/.rvm/rubies/ruby-2.5.1/lib/ruby/2.5.0/net/http/response.rb:43

retained objects by class
------------------------------------
       119  String
```

History

#1 - 09/06/2018 09:45 AM - chopraanmol1 (Anmol Chopra)

It can be an issue related to MemoryProfiler or some edge case to ObjectSpace.trace_object_allocations_start

The following code also result in a similar result:

```
MemoryProfiler.report{ 200.times{|i| "SOME RANDOM TEXT#{i}: SOME Value".dup } }.pretty_print
```

Note if you don't see the same result try increasing string length.

I've tried doing same without MemoryProfiler in an infinite loop and it doesn't increase memory size

#2 - 09/06/2018 11:28 AM - chopraanmol1 (Anmol Chopra)

Upon further inspection, it seems duplicating long text with interpolation allocates an extra string(frozen). You can observe this using the following script.

```
def string_info
  GC.disable
  obj_ids = []
  final_ids = []
  obj_sps = ObjectSpace.each_object(String)
  obj_sps.each{|i|obj_ids << i.__id__ }
  obj_ids << obj_ids.__id__
  yield
  obj_sps.each{|i|final_ids << i.__id__ }
  ref =  (final_ids - obj_ids).collect{|i| ObjectSpace._id2ref(i)}
  GC.enable
  ref
end

a = Array.new(20){"SOME RANDOM LONG TEXT WITH INTERPOLATION#{}"} # long string + interpolation
p string_info{ a.each(&:dup) }.count # creates two string
a = Array.new(20){"SOME RANDOM LONG TEXT WITHOUT INTERPOLATION"} # long string
p string_info{ a.each(&:dup) }.count # creates one string
a = Array.new(20){"WITH#{} INTERPOLATION"} # small string + interpolation
p string_info{ a.each(&:dup) }.count # creates one string
```

It doesn't seem like a memory leak issue. But it can be argued whether an extra frozen string should be allocated for same content multiple time during duplication of string.

#3 - 09/11/2018 05:16 AM - chopraanmol1 (Anmol Chopra)

alexis (Alexis Bernard) Look into this PR https://github.com/SamSaffron/memory_profiler/pull/59

Once merged this PR should fix the issue you faced.

I think this can be closed now.

#4 - 09/11/2018 03:12 PM - alexis (Alexis Bernard)

Thanks for pointing this PR. I ran it again with this specific memory_profiler version and there is no memory leaks.

It seems that I don't have enough permissions to close the issue.

#5 - 09/17/2018 07:01 PM - alexis (Alexis Bernard)

I kept to investigate the memory leak issue. It seems it comes when OpenSSL::SSL::VERIFY_PEER is enabled and a ca_file is specified :

```
require "net/http"
require "openssl"

def repeat_https_get
  uri = URI("https://example.com/")
  http = Net::HTTP.new(uri.host, uri.port)
  http.verify_mode = OpenSSL::SSL::VERIFY_PEER
  http.ca_file = "certdata.pem" # Download https://curl.haxx.se/ca/cacert.pem
  http.use_ssl = true

  loop { http.request(Net::HTTP::Get.new(uri.path)) }
end

repeat_https_get
```

The VmRSS of the Ruby process grows endlessly. Unfortunately MemoryProfiler does not report any retained memory.