Redmine - Defect #16353

Regexp bug in JournalsController regexp handling when quoting existing journal entries

2014-03-14 11:12 - Stephane Lapie

Status:	Closed	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:	Jean-Philippe Lang	**% Done:**	0%
Category:	Issues	**Estimated time:**	0.00 hour
Target version:	2.5.1		
Resolution:	Fixed	**Affected version:**	2.3.1

Description

I have stumbled upon a very nasty bug with journal entries quoting, when a user writes an entry with a "pre" tag but forgets to put a proper "/pre" tag.

When trying to quote such an entry where a "pre" tag is not closed properly, the regexp engine can go haywire (process stuck in gsub() forever, CPU usage 100%, if it happens with all threads of an existing instance, the whole Redmine instance becomes unusable until the process finishes/is terminated) with cases where a lot of whitespaces/tabs/newlines are present.

Upon stumbling on that bug, and killing the affected Rails processes, I got the following trace :

```
Completed 500 Internal Server Error in 7016997ms
SignalException (SIGTERM):
  app/controllers/journals_controller.rb:69:in `gsub'
  app/controllers/journals_controller.rb:69:in `new'
```

Which allowed me to narrow it down the code where "pre" blocks are replaced with "[...]".

Studying the ruby regular expression spec (http://www.tutorialspoint.com/ruby/ruby_regular_expressions.htm) it seems the following expression `%r{<pre>((.|\s)*?)</pre>}m`
has redundant elements : a "dot" is supposed to match anything including newlines, when the regexp has the 'm' flag active, so there is no purpose in having a "or" match with whitespaces.

I tried the following fix on my side (basically, removing the redundant "or \s" check and sticking with "." and the 'm' flag) :

```
--- /usr/local/share/redmine/app/controllers/journals_controller.rb    2013-05-02 00:15:29.000000000 +0900
+++ /tmp/journals_controller.rb    2014-03-14 18:35:24.498448622 +0900
@@ -66,7 +66,7 @@
       text = @issue.description
     end
     # Replaces pre blocks with [...]
-    text = text.to_s.strip.gsub(%r{&lt;pre&gt;((.|\s)*?)&lt;/pre&gt;}m, '[...]')
+    text = text.to_s.strip.gsub(%r{&lt;pre&gt;(.*?)&lt;/pre&gt;}m, '[...]')
     @content = "#{ll(Setting.default_language, :text_user_wrote, user)}\n&gt; "
     @content &lt;&lt; text.gsub(/(\r?\n|\r\n?)/, "\n&gt; ") + "\n\n"
   rescue ActiveRecord::RecordNotFound
```

and it seems to fix the issue.

How to reproduce/check (it "works" in every case, as in, it WILL eventually end, as it is not an infinite loop, but the execution time varies greatly) :

This works (obviously) :

```
$ time ruby -e 'print "<pre>  lalala       \tlalala</pre>".gsub(%r{<pre>((.|\s)*?)</pre>}m, '"[...]"')'
[...]
real    0m0.015s
user    0m0.011s
sys    0m0.004s
$ time ruby -e 'print "<pre>  lalala       \tlalala\n     lalala\r\n   lala\r    \n        lalalla \t\n</pre>".gsub(%r{<pre>((.|\s)*?)</pre>}m,
"[...]")'
[...]
real    0m0.018s
user    0m0.014s
sys    0m0.005s
```

I apologize in advance for the display break, but there is no way I can provide proper sample code for this problem otherwise... (I hope this does not trigger anything weird by writing it in the description, if this has not been fixed in latest versions...)

These take more and more time as you add more characters, rendering a whole running ruby process unusable :

```
# Starting with 20 characters, mixing spaces and alphabetic
$ time ruby -e 'print "<pre>  lalala       ".gsub(%r{<pre>((.|\s)*?)</pre>}m, '"[...]"')'
<pre>  lalala
real    0m0.069s
user    0m0.007s
sys    0m0.004s
# Adding 7 more characters (total : 27)
$ time ruby -e 'print "<pre>  lalala       \tlalalal".gsub(%r{<pre>((.|\s)*?)</pre>}m, '"[...]"')'
<pre>  lalala          lalalal
real    0m0.018s
user    0m0.013s
sys    0m0.008s
# Adding 14 more characters (total : 41)
$ time ruby -e 'print "<pre>  lalala       \tlalala\n     lalala\r\n".gsub(%r{<pre>((.|\s)*?)</pre>}m, '"[...]"')'
<pre>  lalala          lalala
    lalala

real    0m0.080s
user    0m0.080s
sys    0m0.001s
# Adding 3 more characters (total : 44)
$ time ruby -e 'print "<pre>  lalala       \tlalala\n     lalala\r\n   ".gsub(%r{<pre>((.|\s)*?)</pre>}m, '"[...]"')'
<pre>  lalala          lalala
    lalala

real    0m0.349s
user    0m0.344s
sys    0m0.004s
# Adding 4 more characters (total : 48)
$ time ruby -e 'print "<pre>  lalala       \tlalala\n     lalala\r\n   lala".gsub(%r{<pre>((.|\s)*?)</pre>}m, '"[...]"')'
<pre>  lalala          lalala
    lalala
```

```
   lala
real    0m1.010s
user    0m1.009s
sys    0m0.001s
# Adding 5 more characters (total : 53)
$ time ruby -e 'print "<pre>  lalala      \tlalala\n     lalala\r\n   lala\r    ".gsub(%r{<pre>((.|\s)*?)</pre>}m, ""[...]"")'
<pre>  lalala          lalala
     lalala
    ala
real    0m5.213s
user    0m5.208s
sys    0m0.001s
# Adding 5 more characters (total : 58)
$ time ruby -e 'print "<pre>  lalala      \tlalala\n     lalala\r\n   lala\r    \n    ".gsub(%r{<pre>((.|\s)*?)</pre>}m, ""[...]"")'
<pre>  lalala          lalala
     lalala
    ala

real    2m36.084s
user    2m36.021s
sys    0m0.028s
</pre>
...And it only gets worse from there.

This works, and ends instantly :
<pre>
$ time ruby -e 'print "<pre>  lalala      \tlalala\n     lalala\r\n   lala\r    \n       lalalla \t\n".gsub(%r{<pre>(.*?)</pre>}m, ""[...]"")'
<pre>  lalala          lalala
     lalala
    ala
       lalalla

real    0m0.014s
user    0m0.014s
sys    0m0.001s
$ time ruby -e 'print "<pre>  lalala      \tlalala\n     lalala\r\n   lala\r    \n       lalalla \t\n</pre>".gsub(%r{<pre>(.*?)</pre>}m,
""[...]"")'
[...]
real    0m0.018s
user    0m0.014s
sys    0m0.004s
</pre>
```

Associated revisions

Revision 12969 - 2014-03-15 11:36 - Jean-Philippe Lang

Fixed flawed regexp for removing pre blocks when quoting notes (#16353).

Patch by Stephane Lapie.

Revision 12973 - 2014-03-17 08:49 - Jean-Philippe Lang

Fixed flawed regexp for removing pre blocks when quoting messages (#16353).

Revision 13026 - 2014-03-29 15:41 - Jean-Philippe Lang

Merged r12969 and r12973 (#16353).

History

#1 - 2014-03-15 11:38 - Jean-Philippe Lang

- Status changed from New to Resolved

- Assignee set to Jean-Philippe Lang

- Target version set to 2.5.1

- Resolution set to Fixed

I guess you're using ruby1.8, right? ruby1.9 doesn't seem to suffer from this regexp issue. The regexp was flawed anyway and your patch is applied in r12969, thanks for pointing this out.

#2 - 2014-03-15 12:00 - Stephane Lapie

Jean-Philippe Lang wrote:

>
> *I guess you're using ruby1.8, right? ruby1.9 doesn't seem to suffer from this regexp issue. The regexp was flawed anyway and your patch is applied in r12969, thanks for pointing this out.*

Thank you.

I confirmed my environment for the sake of exhaustivity, if this can help someone pinpoint issues :
- Ruby version : I do have 1.8 and 1.9 installed on this machine, but I can certify that Redmine uses 1.9 (as follows)
- Web server : Apache + mod_passenger

The configuration for mod_passenger on my Redmine server, and the used version of Ruby are as follows :

```
$ cat /etc/apache2/mods-enabled/passenger.conf
<IfModule mod_passenger.c>
  PassengerRoot /usr
  PassengerRuby /usr/bin/ruby
</IfModule>
$ /usr/bin/ruby -v
ruby 1.9.3p194 (2012-04-20 revision 35410) [x86_64-linux]
$ /usr/bin/ruby1.9.1 -v
ruby 1.9.3p194 (2012-04-20 revision 35410) [x86_64-linux]
$ /usr/bin/ruby1.9.3 -v
ruby 1.9.3p194 (2012-04-20 revision 35410) [x86_64-linux]
$ ps auxw | grep Rack
www-data 11471  0.2  3.3 413872 136696 ?       Sl   15:43   0:35 Rack: /usr/local/share/redmine
$ lsof -p 11471 | grep -E "ruby1|ruby-1"
ruby    11471 www-data  txt    REG              254,0     6336    22798 /usr/bin/ruby1.9.1
ruby    11471 www-data  mem    REG              254,0  2072592    24013 /usr/lib/libruby-1.9.1.so.1.9.1
```

I also find it extremely odd that Debian's ruby packages install stuff this way, for what it's worth. I suspect in this instance that the problem got fixed in a further version of ruby 1.9, and that I am maybe using an older unfixed version.

#3 - 2014-03-17 03:08 - Stephane Lapie

Confirming the same problem in app/controllers/messages_controller.rb at line 116 :

```
--- app/controllers/messages_controller.rb    2014-02-08 17:19:32.000000000 +0900
+++ /tmp/messages_controller.rb    2014-03-17 11:07:46.070637759 +0900
@@ -113,7 +113,7 @@
     @subject = "RE: #{@subject}" unless @subject.starts_with?('RE:')

     @content = "#{ll(Setting.default_language, :text_user_wrote, @message.author)}\n> "
-    @content << @message.content.to_s.strip.gsub(%r{&lt;pre&gt;((.|\s)*?)&lt;/pre&gt;}m, '[...]').gsub(/(\r?\n|\r\n?)/, "\n&gt; ") + "\n\n"
+    @content &lt;&lt; @message.content.to_s.strip.gsub(%r{&lt;pre&gt;(.*?)&lt;/pre&gt;}m, '[...]').gsub(/(\r?\n|\r\n?)/, "\n&gt; ") + "\n\n"
   end

   def preview
```

#4 - 2014-03-29 15:41 - Jean-Philippe Lang

- Status changed from Resolved to Closed

Merged.