# ChiliProject - Bug # 934: @project.repository.tags returns StringIO object if tags are not cached

| Status: | Declined | Priority: | Normal |
|---|---|---|---|
| Author: | Florian Pommerening | Category: | |
| Created: | 2012-03-16 | Assignee: | |
| Updated: | 2012-03-23 | Due date: | |

| **Remote issue URL:** | | | |
|---|---|---|---|
| **Affected version:** | 3.0.0 | | |
| **Description:** | This happens in git repositories with the redmine gitolite plugin installed, but looking at the code, I guess it could happen with git repositories in general. |

```
>> p = Project.find("myproject")
=> #<Project id: 300, name: "myproject", description: "", homepage: "", is_public: false, parent_id: 285,
created_on: "2012-02-27 13:51:22", updated_on: "2012-02-27 13:51:22", identifier: "myproject", status: 1, lft:
493, rgt: 494>
>> r = p.repository
=> #<Repository::Git id: 310, project_id: 300, url: "repositories/myproject.git", login: "", password: "", root_url:
"repositories/myproject.git", type: "Repository::Git", path_encoding: nil, log_encoding: nil>
>> t = r.tags
=> #<StringIO:0xb5384d60>
>> t.eof?
=> true
>> t = r.tags
=> ["deletethistag", "deletethistag2"]
```

One easy fix is to change
```
 scm_cmd(*cmd_args) do |io|
   @tags = io.readlines.sort!.map{|t| t.strip}
 end
```

into

```
 scm_cmd(*cmd_args) do |io|
   @tags = io.readlines.sort!.map{|t| t.strip}
 end
 @tags
```

in the method tags of GitAdapter.

## History

**2012-03-20 09:18 pm - Felix Schäfer**

Hello Florian,

Would you be able to write a test for this?

**2012-03-21 09:31 am - Florian Pommerening**

Hello Felix,

I'm new to Ruby/Rails and I'm not sure how to write a test. I looked at the existing tests and saw test/unit/repository_git_test.rb It looks like this uses a repository that already contains tags (well, at least one). I guess the following lines added to this file would be enough:

<pre>
  def test_tags
    tags = @repository.tags
    assert_equal [
        'tag01.annotated',
      ], tags
  end
</pre>

However, due to the caching, this test would only fail if it is the first time the method tags is called. If tags was called on the same repository before, it will return the correct result cached in its @tags instance variable.

**2012-03-21 07:10 pm - Felix Schäfer**

I can't reproduce that:

<pre>
>> r = Project.first.repository
=> #<Repository::Git id: 1, project_id: 1, url: "/Users/felix/Code/tmp/grack/.git", login: "", password: "", root_url: "/Users/felix/Code/tmp/grack/.git", type: "Repository::Git", path_encoding: "", log_encoding: nil>
>> r.tags
=> ["v0.1", "v0.2"]
</pre>

Can you confirm this is a problem on a vanilla installation too or if it's only if you have the gitolite plugin installed?

**2012-03-21 07:10 pm - Felix Schäfer**
*- Status changed from Open to Needs more information*

**2012-03-22 09:32 am - Florian Pommerening**
*- Status changed from Needs more information to Ready for review*

I'm sorry, I cannot test on a vanilla instance right now. Maybe it has something to do with the ruby version (1.8.7)? Could be related to this "bug in ruby":http://bugs.ruby-lang.org/issues/159

**2012-03-22 10:59 am - Felix Schäfer**
*- Status changed from Ready for review to Needs more information*

(Please don't set the status of this to "Ready for review" until the scope and causes of this have been fully studied, thanks)

I couldn't reproduce this on ruby 1.8.7 (2011-06-30 patchlevel 352), so it's not related to any "recent" version of 1.8.7 (the linked bug is 4 years oldâ€¦), what exact version of ruby are you on?

**2012-03-22 12:05 pm - Florian Pommerening**
<pre>
>> ruby -v
ruby 1.8.7 (2011-06-30 patchlevel 352) [i686-linux]

</pre>

So its probably not the ruby version :-)

I just did another test and the issue doesn't occur with our development instance (which doesn't use the gitolite plugin and is set to the environment "development"). It still happens with our live system, though (gitolite plugin and set to the "production" rails environment). Unfortunately, I cannot test whether it is the production environment or the gitolite plugin that makes the difference.

Another guess:
vendor/plugins/redmine_git_hosting/lib/git_hosting/patches/git_adapter_patch.rb line 97 returns the io object which seems to me like it could cause this issue (keep in mind that I'm just starting out with ruby, so this is really just a guess)

**2012-03-22 04:11 pm - Felix Schäfer**
Florian Pommerening wrote:
> Another guess:
> vendor/plugins/redmine_git_hosting/lib/git_hosting/patches/git_adapter_patch.rb line 97 returns the io object which seems to me like it could cause this issue (keep in mind that I'm just starting out with ruby, so this is really just a guess)

This looks like a plugin, if you give me a link to the source I can have a look.

**2012-03-22 06:44 pm - Florian Pommerening**
Yes, this is the "gitolite plugin":https://github.com/ericpaulbishop/redmine_git_hosting I was talking about. Sorry, I should have mentioned that from the start. Here is the "line of code I
mentioned":https://github.com/ericpaulbishop/redmine_git_hosting/blob/master/lib/git_hosting/patches/git_adapter_patch.rb#L97

**2012-03-22 08:05 pm - Felix Schäfer**
This looks like the plugin is the culprit. Unless you can confirm this also happens on ChiliProject without the plugin, I'll have to decline this issue and ask you to contact the plugin author.

**2012-03-23 10:32 am - Florian Pommerening**
Ok, I'll do that. Thanks for your help.

**2012-03-23 10:54 am - Florian Pommerening**
For posterity's sake here is link to the "created issue":https://github.com/ericpaulbishop/redmine_git_hosting/issues/156

**2012-03-23 10:58 am - Felix Schäfer**
*- Status changed from Needs more information to Declined*

Closing here, thanks a lot for the feedback.