

## Redmine - Defect #6001

### Error on login/logout

2010-07-30 08:51 - Max Meier

|                        |            |                          |            |
|------------------------|------------|--------------------------|------------|
| <b>Status:</b>         | Closed     | <b>Start date:</b>       | 2010-07-30 |
| <b>Priority:</b>       | Normal     | <b>Due date:</b>         |            |
| <b>Assignee:</b>       |            | <b>% Done:</b>           | 0%         |
| <b>Category:</b>       | Plugin API | <b>Estimated time:</b>   | 0.00 hour  |
| <b>Target version:</b> |            | <b>Affected version:</b> |            |
| <b>Resolution:</b>     | Invalid    |                          |            |

#### Description

I'm running Redmine with two environments, production and development. When I try to log in or log out on production, everything is fine. When I do the same on development instance, I get

```
ArgumentError in AccountController#logout
```

```
A copy of ApplicationController has been removed from the module tree but is still active!
```

```
/usr/lib/ruby/gems/1.8/gems/activesupport-2.3.5/lib/active_support/dependencies.rb:414:in `load_missing_constant'
```

```
/usr/lib/ruby/gems/1.8/gems/activesupport-2.3.5/lib/active_support/dependencies.rb:96:in `const_missing'
```

```
/var/lib/redmine.stable/app/controllers/application_controller.rb:54:in `user_setup'@
```

I'm running 0.9-stable-[r3898](#).

Here's the output from RAILS\_ENV=development script/about:

```
/usr/lib/ruby/gems/1.8/gems/rails-2.3.5/lib/rails/gem_dependency.rb:119:Warning: Gem::Dependency#version_requirements is deprecated and will be removed on or after August 2010. Use #requirement
```

About your application's environment

```
Ruby version          1.8.7 (i686-linux)
RubyGems version      1.3.6
Rack version          1.0
Rails version         2.3.5
Active Record version 2.3.5
Active Resource version 2.3.5
Action Mailer version 2.3.5
Active Support version 2.3.5
Application root      /var/lib/redmine.stable
Environment            development
Database adapter      mysql
Database schema version 20100313171051
```

About your Redmine plugins

```
Redmine Bugcloud plugin      0.0.2.1
Scrumdashboard plugin        1.2
Redmine Backlogs plugin      0.0.1
Redmine Attach Screenshot plugin 0.0.3
```

#### History

##### #1 - 2010-07-30 09:31 - Felix Schäfer

I'm not familiar with all the plugins, but I'd suppose one of them is cross-firing somewhere.

##### #2 - 2010-07-30 09:39 - Max Meier

Hm, I don't think so, because the other instance of redmine (production) has the same plugins activated and works well.

Here's the info for production environment:

```
RAILS_ENV=production script/about
/usr/lib/ruby/gems/1.8/gems/rails-2.3.5/lib/rails/gem_dependency.rb:119:Warning: Gem::Dependency#version_requirements is deprecated and will be removed on or after August 2010. Use #requirement
About your application's environment
Ruby version          1.8.7 (i686-linux)
RubyGems version      1.3.6
Rack version          1.0
Rails version         2.3.5
Active Record version 2.3.5
Active Resource version 2.3.5
Action Mailer version 2.3.5
Active Support version 2.3.5
Application root      /var/lib/redmine.stable
Environment           production
Database adapter      mysql
Database schema version 20100313171051
```

```
About your Redmine plugins
Redmine Bugcloud plugin 0.0.2.1
Redmine Attach Screenshot plugin 0.0.3
Redmine Backlogs plugin 0.0.1
Scrumdashboard plugin 1.2
```

### #3 - 2010-07-30 13:08 - Jean-Baptiste Barth

- *Category set to Plugin API*
- *Status changed from New to Closed*
- *Resolution set to Invalid*

It's a plugin bug. It occurs when you run a plugin in development environment and this plugin uses a "require" instead of a "require\_dependency" somewhere. Plus "require" should be put in a "config.to\_prepare" or "Dispatcher.to\_prepare" block in init.rb. Behind this, this plugin (I don't know which one..) try to force reload a file which already exists in memory.

I close this bug since it does not concern the core. I advise you to disable plugins one by one and identify the one which causes this bug, then file a bug on the tracker of this plugin if any.

### #4 - 2010-07-30 13:11 - Jean-Baptiste Barth

Just one more precision : it only happens in development environment because when you are in a production environment, everything is loaded once, I mean Rails never tries to reload a class from files if it has already loaded it before. Hence the difference you observed..