Redmine - Patch #9317

Admin users should be always authorized when no context is given

2011-09-25 10:52 - Alex Shulgin

Status:	New	**Start date:**	2011-09-25
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:	Permissions and roles	**Estimated time:**	0.00 hour
Target version:			

Description

In a situation where @project might or might not be set by filters before the call to authorize you would need to check for that and either call authorize or authorize_global (otherwise admin user will get '403 You are not authorized to access this page', which is ridiculous).

With this patch applied, a plain before_filter :authorize may be used instead.

History

#1 - 2012-07-24 16:33 - Alex Shulgin

Hello,

I'd like to revisit this issue.

The more I think about it the less sense a notion of 'authorize_global' makes to me. Why a permission to create new projects should be granted to a role? A role only makes sense in the context of user membership in a project. To me, the whole system of 'global authorization' looks like a bad attempt to fool it around.

Why the manager of 'A tiny and unimportant project' should be granted permission to pollute the top-level projects scope? Why a permission do anything on that unimportant project will magically grant any member of that project the permission to do the same thing 'globally' (whatever that really means?)

I believe we should grant the permission to create new top-level projects **to admins only,** but let project managers create *subprojects* instead (i.e., add new permission 'Create subprojects'.)

I suggest removing the 'authorize_global' altogether, along with handling of ':global => true' parameter in 'User.allowed_to?'. If no context is given, only admin users should be authorized, as originally suggested in this ticket.

I can count only 3 usages of 'authorize_global' in the current core code:

```
projects_controller.rb:  before_filter :authorize_global, :only => [:new, :create]
```

This is covered above: add 'create subproject' permission.

```
timelog_controller.rb:  before_filter :authorize_global, :only => [:new, :index, :report]
```

Here, effectively we need to query every project where the current user is a member to see if he is authorized to log spent time and list only such projects in the drop down (on new action.) Ditto for index/report, but check different permission ('view spent time'.) We can just code that explicitly.

```
attachments_controller.rb:  before_filter :authorize_global, :only => :upload
```

I failed to locate where exactly this action is used, but I believe it can be made to work w/o the need for global authorization.

I'd be happy to work on this (submit more patches,) I just want to make sure this will have any sort of support from upstream (or fellow Redmine users.)

--
Kind regards,
Alex

#2 - 2012-07-24 20:41 - Alex Shulgin

What I forgot to mention is how I've came over this issue.

I've written this plugin[1] where you can store prepared issue responses per-project or globally. The problem is that an admin user gets error 403 if he visits the global canned responses URI. That is totally unacceptable: admin should never get 'access forbidden' error message.

To overcome the problem one might need to add admin to at least one project **and** use the infamous 'autorize_global' method. That doesn't make much sense to me, so here I am trying to fix this in Redmine core. Please bear with me ;-)

[1] https://github.com/commandprompt/redmine_canned_responses

#3 - 2012-07-24 21:14 - Jean-Philippe Lang

> I believe we should grant the permission to create new top-level projects to admins only, but let project managers create subprojects instead (i.e., add new permission 'Create subprojects'.)

I don't know which version you're using but the 'Create subprojects' permission was added a while ago.

#4 - 2012-07-24 21:21 - Alex Shulgin

Jean-Philippe Lang wrote:

> I don't know which version you're using but the 'Create subprojects' permission was added a while ago.

Bummer, I've just upgraded from 1.3 to 1.4. Now I see it was in 1.3 as well. Anyway, I suggest removing the original 'Create project' permission in favor of leaving it to admin users only.

#5 - 2013-03-28 01:03 - Anonymous

Hmm, indeed, it seems odd to allow the manager of a project to create new top-level projects... Might be OK to leave this setting in, but it should be disabled by default for all roles, shouldn't it?

Also, I wonder about the patch that's attached here. Precisely because I stumbled across the same issue: Using a plugin that adds a new permission, and not remembering to turn it on for all kinds of roles, leads to the admin user not being able to exercise the ability controlled by that new permission. That seems odd -- admins should be able to do anything and everything, shouldn't they?

#6 - 2013-03-28 13:34 - Anonymous

On a somewhat tangential note, there is also a function allowed_to_globally define in app/models/user.rb -- however, nothing uses it, indeed its name occurs only once in the complete redmine sources. It was added in SVN revision 4164 by Eric Davis (Sep 20, 2010). Perhaps some plugins use it?

#7 - 2013-03-28 14:17 - Anonymous

To further elaborate on what Alex already explained: Even if one wants to use :global=>true, this fails as soon as one wants to use link_to_if_authorize() -- there seems to be no way to tell that to pass :global=>true. So one ends up with evil hacks like the on I put here for Alex' canned response plugin: https://github.com/commandprompt/redmine_canned_responses/pull/16

If there was at least some sign of interest by the Redmine devs, that would be encouraging :-).

Files

0001-Admin-user-is-always-authorized-when-no-context-is-g.patch	954 Bytes	2011-09-25	Alex Shulgin