Redmine - Feature #16309

Add a concept of role-based permission to trackers

2014-03-11 10:42 - Stephane Lapie

Status:	Closed		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:			**% Done:**	0%
Category:	Permissions and roles		**Estimated time:**	0.00 hour
Target version:				
Resolution:	Duplicate			

Description

I have a need for limiting access to trackers based on roles. For instance:
- only a Developper or Manager role could report a new "Dev Request" issue but not an "Infra Change"
- but only a Sysadmin or Manager role could report an "Infra Change" issue but not a "Dev Request"

So far, there is a plugin (http://www.redmine.org/plugins/redmine_track_control) that seemed to fit that need, except it was only constraining the tracker choice via UI, and had a few other problems.

I made my own fork (https://github.com/darksoul42/redmine_track_control) in which I tried fixing them on my own :
- creating a permission symbol based on tracker ID instead of potentially non-ASCII tracker name (this would break since ActiveRecord could not recover symbol names from the database, when they had Japanese in them)
- handling the display of my home-cooked "create_tracker#{id}" as "Create #{name} tracker" with dynamically added translations upon displaying Roles
- actually enforce tracker validity check at every level (override on IssuesController), instead of just making a JavaScript popup alert (still needing an override on Issues views to limit which trackers will be displayed)

But I stumbled upon the following problem :
- Basically, I am overriding IssuesController and more specifically, build_new_issue_from_params and update_issue_from_params. In order to do stuff properly in regard to handling custom fields (which require the tracker be set beforehand), this would require me to completely rewrite the whole method, just to change this one line in the middle, with quite the ugly code block :

```
-    @issue.tracker ||= @project.trackers.find((params[:issue] && params[:issue][:tracker_id]) ||
params[:tracker_id] || :first)
+    if project.enabled_modules.where(:name => "tracker_permissions").count == 1
+      tracker_list = @project.trackers.select { |t| User.current.allowed_to?("create_tracker#{t.i
d}".to_sym, @project, :global => true) }
+    else
+      tracker_list = @project.trackers
+    end
+    @issue.tracker ||= tracker_list.find((params[:issue] && params[:issue][:tracker_id]) || param
s[:tracker_id] || :first)
```

It doesn't sound realistic to enforce a in-house patch to core functionality (be it the Tracker functionality or the Issues functionality) like this, then try to play catch-up with Redmine everytime there is a new version (in addition to the fact the code seems to be about to be refactored and otherwise radically modified).

The above is my rationale for why this should not be a separate plug-in or a patch, but a core feature.

I have the backing of my company for contributing, if there is anything I can do to help implement this in Redmine core.

Related issues:

Is duplicate of Redmine - Feature #285: Tracker role-based permissioning	**Closed**

History

#1 - 2014-03-12 08:51 - Daniele Pedroni

+1
I'm using the same redmine_track_control plugin but I agree with Stephane: this would be a great core feature!

#2 - 2014-04-09 02:53 - Toshi MARUYAMA

- Description updated

#3 - 2015-06-17 22:59 - Alex Petty

+1 -- role based permissions governing tracker level elements would be a very important core feature which is now sorely missing from Redmine.

#4 - 2019-06-02 06:23 - Go MAEDA

- Is duplicate of Feature #285: Tracker role-based permissioning added

#5 - 2019-06-02 06:24 - Go MAEDA

- Status changed from New to Closed

- Resolution set to Duplicate

The feature was added in Redmine 3.3.0 ([#285](#)).