

Redmine - Defect #18855

User with only Move Issue rights in the project can still create issues using mass copy!

2015-01-15 19:52 - Scott Cunningham

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:	Jean-Philippe Lang	% Done:	0%
Category:	Issues permissions	Estimated time:	0.00 hour
Target version:	3.0.0	Affected version:	2.5.2
Resolution:	Fixed		
Description			
<p>I found this bug when I was trying to use a project with a list of issues as a template for other projects (process flow). I assigned members to custom role "Copy" which only allows viewing and moving issues. If, however, the user does not change the project (i.e. copy into other project), new issues will be created within the existing project where they do not have rights!</p> <p>I am running 2.5.2 on a Bitnami stack. I do not have the chance to try 2.6.x at the moment.</p> <p>Note - we use <i>task</i> instead of <i>issue</i> in our language file.</p>			
Custom Role Copy settings:			
01-role-rights.png			
User does not have issue edit rights (correct)			
02-no-edit-rights.png			
User can copy multiple issues at once (correct)			
03-multiple-copy-possible.png			
Copy screenshot			
04-copy-screen.png			
Issues were added to existing project without regard to no Add Issue rights (not correct)			
05-issues-added-without-rights.png			
Related issues:			
Related to Redmine - Patch # 28311: Remove unused i18n key "permission_move_i...		Closed	

Associated revisions

Revision 13981 - 2015-02-08 11:20 - Jean-Philippe Lang

Removed :move_issues permission (#18855).

This permission was wrongly used to allow bulk issue copy. To prevent user from moving an issue to another project, the project field should now be set to read-only in the workflow permissions. A migration does this automatically for roles that have the edit_issues permission without having the move_issues permission.

Revision 13985 - 2015-02-08 13:07 - Jean-Philippe Lang

Adds a `:copy_issues` permission (#18855).

When copy is allowed, target projects are those on which the user has the `:add_issues` permission.

Revision 13986 - 2015-02-08 13:07 - Jean-Philippe Lang

Adds `:permission_copy_issues` string to locales (#18855).

Revision 13990 - 2015-02-08 16:25 - Jean-Philippe Lang

Fixed the migration for SQLServer (#18855).

History

#1 - 2015-01-15 21:46 - Scott Cunningham

I believe I have tracked down the problem.

Context menu *Copy* calls the `bulk_edit` function in `issues_controller.rb`:

1. checks if user has `move_issue` rights
2. builds an allowed projects list by calling `allowed_target_projects_on_move` in `issue.rb`:
3. which checks projects for `move` rights, not `add` rights...

So for copy, the program checks for move-out and move-in rights. But move-in rights is really add rights.

I think instead, `move` rights should be checked at source project and then `add` rights at destination project. This should block a user from copying issues into a project where they do not have `add_issue` rights.

`issues_controller.rb` snippet

```
# Bulk edit/copy a set of issues
def bulk_edit
  @issues.sort!
  @copy = params[:copy].present?
  @notes = params[:notes]

  if User.current.allowed_to?(:move_issues, @projects) # <----- this is correct: can user move/copy in the first place
    @allowed_projects = Issue.allowed_target_projects_on_move # <----- i think this is wrong: target projects should only be add
rights
  if params[:issue]
    @target_project = @allowed_projects.detect { |p| p.id.to_s == params[:issue][:project_id].to_s }
    if @target_project
      target_projects = [@target_project]
    end
  end
end
end
target_projects ||= @projects
```

#2 - 2015-01-15 22:18 - Scott Cunningham

I made a small patch and destination projects are now only ones with `Add issue` rights.

Unresolved: If the user does not change the project pull down from *(No change)*, then new issues will still be created even when the permissions should not allow it. This is past my knowledge point now.

1. Modify models\issue.rb file:

```
# Returns a scope of projects that user can move issues to
def self.allowed_target_projects_on_move(user=User.current)
  Project.where(Project.allowed_to_condition(user, :move_issues))
end

# Returns a scope of projects that user can add issues to # <--- new
def self.allowed_target_projects_on_copy(user=User.current) # <--- new
  Project.where(Project.allowed_to_condition(user, :add_issues)) # <--- new
end
```

2. Modify controllers\issues_controller.rb file:

```
# Bulk edit/copy a set of issues
def bulk_edit
  @issues.sort!
  @copy = params[:copy].present?
  @notes = params[:notes]

  if User.current.allowed_to?(:move_issues, @projects)
    #@allowed_projects = Issue.allowed_target_projects_on_move # <---- comment out
    @allowed_projects = Issue.allowed_target_projects_on_copy # <---- new line
    if params[:issue]
      @target_project = @allowed_projects.detect {|p| p.id.to_s == params[:issue][:project_id].to_s}
      if @target_project
        target_projects = [@target_project]
      end
    end
  end
  target_projects ||= @projects
end
```

#3 - 2015-01-18 21:46 - Jean-Philippe Lang

- Target version set to Candidate for next major release

#4 - 2015-02-08 15:57 - Jean-Philippe Lang

- Status changed from New to Closed
- Assignee set to Jean-Philippe Lang
- Target version changed from Candidate for next major release to 3.0.0
- Resolution set to Fixed

This is now fixed. The :move_issues permission is removed (r13981) and replaced with a :copy_issues permission (r13985). When allowed to copy issues, use can copy them to projects on which he has the :add_issues permission.

#5 - 2018-03-14 02:55 - Go MAEDA

- Related to Patch #28311: Remove unused i18n key "permission_move_issues" added

Files

01-role-rights.png	20.7 KB	2015-01-15	Scott Cunningham
--------------------	---------	------------	------------------

02-no-edit-rights.png	7.15 KB	2015-01-15	Scott Cunningham
03-multiple-copy-possible.png	7.84 KB	2015-01-15	Scott Cunningham
04-copy-screen.png	24.6 KB	2015-01-15	Scott Cunningham
05-issues-added-without-rights.png	15.1 KB	2015-01-15	Scott Cunningham