

Redmine - Defect #26145

Don't redirect anonymous users to the login form for disabled modules

2017-06-12 23:41 - Adam Clark

| | | | |
|------------------------|-----------------------|--------------------------|-----------|
| Status: | Closed | Start date: | |
| Priority: | Normal | Due date: | |
| Assignee: | Jean-Philippe Lang | % Done: | 0% |
| Category: | Permissions and roles | Estimated time: | 0.00 hour |
| Target version: | 4.0.0 | Affected version: | 2.5.2 |
| Resolution: | Fixed | | |

Description

The use case is that a project enables a module (the Files module is the one we encountered) but later turns it off. If there are stray links to that URL, these generate a 403 error, which for anonymous users redirects to the login/registration page.

In our case, we had projects that served files publicly, then moved these to another location and turned off the Files module. The result is we now get a lot of spurious registration requests from users who are trying to download these files, because to the user it looks like the site is asking them to register before they can access the files.

I was able to address our immediate problem by patching ApplicationController.authorize to check whether the request is associated with a disabled project module, and redirect to the main project page in that case. I'm not sure how correct this code is (I don't know the Redmine internals all that well) and I know that this doesn't work for some modules (eg. Issues) which apparently operate through some other method. So this code is just for illustration, I guess.

```
def authorize_with_custom(ctrl = params[:controller], action = params[:action], global = false)
  allowed = User.current.allowed_to?({:controller => ctrl, :action => action}, @project || @projects, :global => global)
  if allowed
    true
  else
    if @project
      logger.debug 'Handling auth error for ' + ctrl + '/' + action

      # Figure out whether the permission for this path is handled by a module
      project_module = Redmine::AccessControl.permissions.select {|p| p.actions.include?(ctrl + '/' + action)}.first.try(
:project_module)
      logger.debug 'Permission module is ' + project_module.to_s

      # If it is a module, and the module isn't enabled in this project, try to redirect to the main project page
      if project_module && !@project.module_enabled?(project_module)
        can_view_project = User.current.allowed_to?({:controller => :projects, :action => :show}, @project)
        logger.debug 'Can the user view the main project page? ' + can_view_project.to_s
        if can_view_project
          redirect_to project_path(@project)
          return false
        end
      end
    end
    if @project && @project.archived?
      render_403 :message => :notice_not_authorized_archived_project
    else
      deny_access
    end
  end
end
```

Associated revisions

Revision 16726 - 2017-06-26 23:04 - Jean-Philippe Lang

Don't redirect anonymous users to the login form when module is disabled (#26145).

History

#1 - 2017-06-26 23:09 - Jean-Philippe Lang

- *Subject changed from Accessing URL for a disabled module generates a 403 error to Don't redirect anonymous users to the login form for disabled modules*
- *Category changed from Files to Permissions and roles*
- *Status changed from New to Closed*
- *Assignee set to Jean-Philippe Lang*
- *Target version set to 4.1.0*
- *Resolution set to Fixed*

Fixed in r16726, they now get a 403 error without being redirected to the login form.

#2 - 2017-07-08 18:37 - Jean-Philippe Lang

- *Target version changed from 4.1.0 to 4.0.0*

#3 - 2017-07-12 05:51 - Toshi MARUYAMA

- *Description updated*