Redmine - Defect #9566
Redmine.pm considers all projects private when login_required is enabled
2011-11-14 20:26 - Guillaume Perréal

Status: Closed
Priority: Normal
Assignee: Jean-Philippe Lang
Category: SCM extra
Target version: 1.2.3
Resolution: 1.2.2

Description
When the "login_required" setting is enabled, Redmine.pm consider all projects as private, which prevents non-member access to public projects.

Associated revisions
Revision 7808 - 2011-11-14 22:40 - Jean-Philippe Lang

Fixed: Redmine.pm considers all projects private when login_required is enabled (#9566).

Revision 7813 - 2011-11-15 19:02 - Jean-Philippe Lang

Reverted r7808 (#9566).

Revision 7834 - 2011-11-18 18:28 - Jean-Philippe Lang

Fixed: Redmine.pm considers all projects private when login_required is enabled (#9566).

Revision 8119 - 2011-12-07 22:57 - Jean-Philippe Lang

Merged r7834 from trunk (#9566).

History
#1 - 2011-11-14 22:42 - Jean-Philippe Lang
- Status changed from New to Resolved
- Resolution set to Fixed

This should be fixed in r7808. Can you confirm?

#2 - 2011-11-14 23:12 - Jean-Philippe Lang
- Assignee set to Jean-Philippe Lang

#3 - 2011-11-15 00:44 - Guillaume Perréal

With r7808, providing any login/password always allows reading on public projects, even if the user does not exist or the password is wrong.
Indeed and I've just reverted it. The fix needs more refactoring.

The attached patch should handle the access of non members. Feedback is welcome.

I think "cast(projects.is_public as CHAR) IN ('t', '1')" could be handled as "projects.is_public" since it should be a boolean and I do not know if this SQL is allowed with all RDMS. ("t" is just how DBI returns PostgreSQL true value.) I will test it with Mysql and maybe Sqlite. By the way, I am not sure how this change would affect the RedmineDbQuery directive.

It doesn't work with sqlite3.

and I do not know if this SQL is allowed with all RDMS. ("t" is just how DBI returns PostgreSQL true value.) I will test it with Mysql and maybe Sqlite.

It should work with mysql, postgresql and sqlite3.

The members and member_roles are no longer available for additional conditions but I don't think it really makes sense to add custom conditions on these tables.

There is a small typo in the subquery: it selects member_roles.id where it should be member_roles.role_id. Once fixed, it behaves as expected: I tested it with roles manager, reviewer and non-member for reading and writing access.
Fixed patch applied, thanks for the feedback.

#10 - 2011-11-18 18:29 - Jean-Philippe Lang
- File deleted (9566.patch)

#11 - 2011-12-07 22:58 - Jean-Philippe Lang
- Status changed from Resolved to Closed

Merged.