Redmine - Patch #16685

Introduce the request_store gem to hold User.current and prevent data leakage in error messages

2014-04-15 15:21 - Holger Just

Status:	Closed	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	100%
Category:	Accounts / authentication	**Estimated time:**	0.00 hour
Target version:	2.6.0		

Description

Recently, there were several issues where the User.current was not properly initialized and thus the setting from the request before it was used, e.g. #16511 with r13041 which can lead to data disclosure. This issue and all similar ones can be fully circumvented by ensuring that the current user is reset before the request even reaches our rails code.

This patch which was extracted from Planio proposes the introduction of the request_store gem. It adds a middleware to provide a true request-local data store based on Thread.current. This ensures that the current user always needs to be set explicitly and can't be taken from previous requests, even if the logic to setup the user somehow fails or is circumvented due to early error handlers. The fail-safe default is always User.anonymous which should be sufficient even in the face of an auth error.

This patch is against the current master branch on Github.

Related issues:

Related to Redmine - Feature #31911: Update request_store gem to 1.4	**Closed**

Associated revisions

Revision 13110 - 2014-05-01 02:44 - Toshi MARUYAMA

introduce request_store to ensure that the current user doesn't leak across request boundaries (#16685)

Contributed by Holger Just.

Revision 13111 - 2014-05-01 03:45 - Toshi MARUYAMA

Merged r13110 from trunk to 2.5-stable (#16685)

introduce request_store to ensure that the current user doesn't leak across request boundaries.

Contributed by Holger Just.

Revision 13112 - 2014-05-01 03:50 - Toshi MARUYAMA

Gemfile: use double quote for request_store (#16685)

Revision 13113 - 2014-05-01 03:52 - Toshi MARUYAMA

Merged r13112 from trunk to 2.5-stable (#16685)

Gemfile: use double quote for request_store

Revision 13114 - 2014-05-01 03:55 - Toshi MARUYAMA

Merged r13110 and r13112 from trunk to 2.4-stable (#16685)

introduce request_store to ensure that the current user doesn't leak across request boundaries.

Contributed by Holger Just.

Revision 13117 - 2014-05-01 13:44 - Toshi MARUYAMA

Gemfile: upgrade mocha version (#16685)

Revision 13118 - 2014-05-01 14:27 - Toshi MARUYAMA

Merged r13117 from trunk to 2.5-stable (#16685)

Gemfile: upgrade mocha version.

Revision 13119 - 2014-05-01 14:28 - Toshi MARUYAMA

Merged r13117 from trunk to 2.4-stable (#16685)

Gemfile: upgrade mocha version.

Revision 13120 - 2014-05-01 15:37 - Toshi MARUYAMA

2.5-stable: revert r13111 and r13113 (#16685)

Revision 13121 - 2014-05-01 15:39 - Toshi MARUYAMA

2.4-stable: revert r13114 (#16685)

Revision 13181 - 2014-06-12 13:43 - Toshi MARUYAMA

Gemfile: pin request_store version 1.0.5 (#16685)

Many integration tests are broken.

History

#1 - 2014-04-15 16:22 - Toshi MARUYAMA

- Target version set to 2.4.6

#2 - 2014-04-15 16:23 - Toshi MARUYAMA

- Description updated

#3 - 2014-05-01 10:35 - Toshi MARUYAMA

Test fails on drone.io.

https://drone.io/github.com/marutosi/redmine/211
https://drone.io/github.com/marutosi/redmine/212

```
  1) Failure:
test_api_should_trigger_basic_http_auth_with_basic_authorization_header(Redmine::ApiTest::AuthenticationTest)
[/xxxxxxxx/redmine/test/integration/api_test/authentication_test.rb:34]:
Expected response to be a <401>, but was <500>
```

Tests on CI Server pass and I cannot reproduce on my CentOS 6.5.
http://www.redmine.org/builds/index.html

#4 - 2014-05-01 13:39 - Toshi MARUYAMA

On Ubuntu 12.04.4 LTS,
ruby 1.9.3p545 (2014-02-24 revision 45159) [x86_64-linux]

```
  1) Failure:
test_api_should_trigger_basic_http_auth_with_basic_authorization_header(Redmine::ApiTest::AuthenticationTest)
[test/integration/api_test/authentication_test.rb:34]:
Expected response to be a <401>, but was <500>

Mocha::ExpectationError (unexpected invocation: #<AnyInstance:UsersController>.authenticate_with_http_basic()
unsatisfied expectations:
- expected exactly once, not yet invoked: #<AnyInstance:ApplicationController>.authenticate_with_http_basic(an
y_parameters)
):
  app/controllers/application_controller.rb:125:in `find_current_user'
  app/controllers/application_controller.rb:100:in `user_setup'
  test/integration/api_test/authentication_test.rb:33:in `test_api_should_trigger_basic_http_auth_with_basic_a
uthorization_header'
```

#5 - 2014-05-01 14:52 - Etienne Massip

Is it really useful to add a new runtime dependency? Wouldn't a complete test be enough instead? And if jot, could that change be targeted to next major release?

#6 - 2014-05-01 15:07 - Toshi MARUYAMA

Etienne Massip wrote:

> Is it really useful to add a new runtime dependency?

mocha is in "test" group.

> Wouldn't a complete test be enough instead?

It depends web servers, so we cannot add new tests.
https://github.com/steveklabnik/request_store#the-problem

> And if not, could that change be targeted to next major release?

This issue is security fix depends on #16511 with r13041,
so I think it should be targeted to minor release.

#7 - 2014-05-01 15:13 - Etienne Massip

I meant request store, not mocha. This should be handled by RoR, not by a specific dependency.

#8 - 2014-05-01 15:22 - Toshi MARUYAMA

We usually upgrade Rails version for security fix in minor release.
And r13110 does not change application behaviors.

#9 - 2014-05-01 15:27 - Etienne Massip

Sure but this is not a security fix, only a potential fix (which from my very personal pov may be a bit of overkill since it seems that all possibilities of having a wrong user in thread are now prevented).

#10 - 2014-05-01 15:40 - Toshi MARUYAMA

- Target version changed from 2.4.6 to 2.6.0

- % Done changed from 0 to 100

OK. I have reverted 2.4-stable and 2.5-stable revisions.

#11 - 2014-05-01 16:15 - Etienne Massip

Thanks, it was just my personal opinion; it may be safer than adding a new gem which might itself contain some security issue we're not aware of!

#12 - 2014-05-05 18:54 - Holger Just

Etienne: It's way too easy to introduce data leaks when dealing with before filters, either in the core or in Plugins, so this adds an additional safety net.

Also, the gem is actually about 10 lines of code, so not that much of a dependency burden. You could add it into the core if you would be willing to support this code (which I think is pointless here). If you read the code on github, you'll notice that there isn't much room for bugs. Also, the library is well tested.

In my opinion, it adds a rather easy-to-use safty net under some easy to break code paths.

#13 - 2014-06-12 13:37 - Toshi MARUYAMA

request_store 1.0.6 breaks many tests.
http://www.redmine.org/builds/logs/build_trunk_postgresql_ruby-2.0.0-p0_1969.html

```
 1) Failure: test_api_should_accept_switch_user_header_for_admin_user(Redmine::ApiTest::AuthenticationTest)
...
```

#14 - 2014-06-13 13:53 - Holger Just

This is probably caused by this commit. In 1.0.6, the thread store is cleared after a request has finished (and not before a request started as before). That means, that after the middleware was passed on the way up again, User.current is not valid anymore.

The solution for this would be to fix the integration tests to not rely on an "old" User.current. In fact, not having this old value still set after a request is a good thing and solves many potential leakage issues.

#15 - 2014-09-14 13:33 - Jean-Philippe Lang

- Status changed from New to Closed

#16 - 2019-08-16 04:27 - Go MAEDA

- Related to Feature #31911: Update request_store gem to 1.4 added

Files

0001-Introduce-request_store-to-ensure-that-the-current-u.patch	1.25 KB	2014-04-15	Holger Just