Redmine - Defect #16107
ApplicationController mishandles non-Basic authentication information, causing an internal error

2014-02-17 10:26 - Stephane Lapie

Status:	Closed	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:	Jean-Philippe Lang	**% Done:**	0%
Category:	Accounts / authentication	**Estimated time:**	0.00 hour
Target version:	2.5.0		
Resolution:	Fixed	**Affected version:**	

Description

I am currently using a 2.3.1 installation (running on Ruby 1.9.3, with a Postgres database), with the Redmine HTTP Auth plug-in, and a mod_auth_kerb setup based on AD, for authentication.

In some cases, accessing from Internet Explorer provokes the following error log :

```
ActiveRecord::StatementInvalid (PG::Error: ERROR:  invalid byte sequence for encoding "UTF8": 0x82
: SELECT "users".* FROM "users"  WHERE "users"."type" IN ('User', 'AnonymousUser') AND "users"."login" =
'`<82>^Gj^F^F+^F^A^E^E^B <82>^G^0<82>^GZ 00.^F *<86>H<82>÷^R^A^B^B^F   *<86>H<86>÷^R^A^B^B^F
+^F^A^D^A<82>7^B^B^^^F
+^F^A^D^A<82>7^B^B
¢<82>^G$^D<82>^G `<82>^G^\^F    *<86>H<86>÷^R^A^B^B^A'):
  app/models/user.rb:352:in `find_by_login'
  app/models/user.rb:163:in `try_to_login'
  app/controllers/application_controller.rb:113:in `block in find_current_user'
  app/controllers/application_controller.rb:112:in `find_current_user'
  app/controllers/application_controller.rb:87:in `user_setup'
```

I have retraced it succesfully to the find_current_user method in "app/controllers/application_controller.rb" (line 111) :

```
# HTTP Basic, either username/password or API key/random
authenticate_with_http_basic do |username, password|
  user = User.try_to_login(username, password) || User.find_by_api_key(username)
end
```

The "authenticate_with_http_basic" code is provided by the actionpack gem, which will access the Authorization header and cut it to get what it thinks is the username and the password. The problem is, since I am not using Basic auth, but Negotiate auth, it is cutting through Kerberos ticket information, and therefore that the resulting [username, password] values will make no sense whatsoever.

After this, it will try using the username value (a binary blob at this point) to look for the corresponding User object, and it is at this point that the database refuses to process the SQL query (which would fail anyway), resulting in an internal error.

I could fix the code on my side with the following fix to app/models/user.rb :

```
--- /usr/local/share/redmine/app/models/user.rb.orig    2014-02-17 16:36:21.246082730 +0900
+++ /usr/local/share/redmine/app/models/user.rb    2014-02-17 17:06:49.078946687 +0900
@@ -155,8 +155,12 @@

   # Returns the user that matches provided login and password, or nil
   def self.try_to_login(login, password)
```

```
-    login = login.to_s
-    password = password.to_s
+### IN-HOUSE PATCH BY STEPHANE LAPIE <stephane.lapie@asahinet.com> ON 2014/02/17
+# This patch here sanitizes the "login" and "password" information.
+# These are derived from parsing the Authorization request header, and they are valid only for Basic authentication.
+# If this info comes from a Negotiate (Kerberos) authentication attempt, they will contain mangled ticket information.
+    login = login.to_s.encode('UTF-8', :invalid => :replace, :undef => :replace)
+    password = password.to_s.encode('UTF-8', :invalid => :replace, :undef => :replace)

     # Make sure no one can sign in with an empty login or password
     return nil if login.empty? || password.empty?
```

By sanitizing the variables like this, the SQL query goes through properly, and even though it fails, authentication is then handed over to the HTTP Auth plugin (using the name stored in the REMOTE_USER variable).

However, at the core, it sounds to me there is a problem in making the assumption that any HTTP auth attempt is going to be Basic, both in Redmine, and in Rails. It would probably be more sensible to check if we are using Basic auth (just check the first word of request.authorization), and if we are not, to right away consider we don't have valid user identification info as we would otherwise anyway, which is why I also patched app/controllers/application_controller.rb :

```
--- /usr/local/share/redmine/app/controllers/application_controller.rb.orig    2014-02-17 18:21:53.497327785 +0900
+++ /usr/local/share/redmine/app/controllers/application_controller.rb    2014-02-17 18:21:32.605134171 +0900
@@ -109,8 +109,12 @@
       user = User.find_by_api_key(key)
     else
       # HTTP Basic, either username/password or API key/random
-      authenticate_with_http_basic do |username, password|
-        user = User.try_to_login(username, password) || User.find_by_api_key(username)
+      ### IN-HOUSE PATCH BY STEPHANE LAPIE <stephane.lapie@asahinet.com> ON 2014/02/17
+      # This patch ensures we don't try Basic authentication in cases where we use any other mechanism
+      if (request.authorization.split(' ') == "Basic")
+        authenticate_with_http_basic do |username, password|
+          user = User.try_to_login(username, password) || User.find_by_api_key(username)
+        end
       end
     end
     # Switch user if requested by an admin user
```

Associated revisions

Revision 12915 - 2014-02-22 13:09 - Jean-Philippe Lang

Trigger basic HTTP authentication only when Basic authorization header is present (#16107).

Revision 12916 - 2014-02-22 13:50 - Jean-Philippe Lang

Strip invalid UTF-8 bytes in User#find_by_login (#16107).

Revision 12917 - 2014-02-22 16:55 - Toshi MARUYAMA

add "assert_response 401" to tests (#16107)

Revision 12918 - 2014-02-22 16:55 - Toshi MARUYAMA

explicitly set encoding UTF-8 (#16107)

Default Ruby source file encoding changed in Ruby 2.0.0.
https://bugs.ruby-lang.org/issues/6679

Revision 12923 - 2014-02-23 09:20 - Jean-Philippe Lang

Merged r12915 to 12918 (#16107).

History

#1 - 2014-02-22 13:56 - Jean-Philippe Lang

- Tracker changed from Patch to Defect

- Subject changed from Redmine's application controller mishandles non-Basic authentication information, causing an internal error to ApplicationController mishandles non-Basic authentication information, causing an internal error

- Status changed from New to Resolved

- Assignee set to Jean-Philippe Lang

- Target version set to 2.5.0

- Resolution set to Fixed

This should be fixed by r12915 and r12916, thanks for pointing this out.

#2 - 2014-02-23 09:20 - Jean-Philippe Lang

- Status changed from Resolved to Closed

Merged.