

## Redmine - Defect #34786

### Missing setting values after upgrade to 4.1.1

2021-02-22 11:59 - Michael Fuchs

<b>Status:</b>	Closed	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	0%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>		<b>Affected version:</b>	4.1.1
<b>Resolution:</b>	Invalid		

#### Description

Trying to upgrade from redmine 4.0.4 to 4.1.1 I get "internal error"s on accessing some pages. In the production.log I find messages involving "undefined method" on setting entries from the DB.

I found problems so far with following values:

- **password\_required\_char\_classes** on /settings:

2021-02-22 09:39:45 +0100 (1) Completed 500 Internal Server Error in 114ms (ActiveRecord: 6.9ms)

2021-02-22 09:39:45 +0100 (1)

2021-02-22 09:39:45 +0100 (1) ActionView::Template::Error (undefined method `password\_required\_char\_classes' for #<Class:0x0000565246162c38>):

2021-02-22 09:39:45 +0100 (1) 20:

21: <p><%= setting\_text\_field :password\_min\_length, :size => 6 %></p>

22:

23: <p><%= setting\_multiselect :password\_required\_char\_classes, Setting::PASSWORD\_CHAR\_CLASSES.keys.collect {|c| [I("label\_password\_char\_class\_#{c}"), c]} , :inline => true %></p>

24:

25: <p>

26: <%= setting\_select :password\_max\_age, [I(:label\_disabled), 0]] + [7, 30, 60, 90, 180, 365].collect{|days| [I('datetime.distance\_in\_words.x\_days', :count => days), days.to\_s]} %>

2021-02-22 09:39:45 +0100 (1)

2021-02-22 09:39:45 +0100 (1) app/helpers/settings\_helper.rb:51:in `setting\_value'

app/helpers/settings\_helper.rb:65:in `setting\_multiselect'

app/views/settings/\_authentication.html.erb:23:in `block in

\_app\_views\_settings\_authentication\_html\_erb\_\_\_3679154722958698456\_70053300436000'

app/views/settings/\_authentication.html.erb:1:in

`\_app\_views\_settings\_authentication\_html\_erb\_\_\_3679154722958698456\_70053300436000'

app/views/common/\_tabs.html.erb:21:in `block in

\_app\_views\_common\_tabs\_html\_erb\_\_\_2546108985375492445\_47455707435340'

app/views/common/\_tabs.html.erb:20:in `each'

app/views/common/\_tabs.html.erb:20:in `\_app\_views\_common\_tabs\_html\_erb\_\_\_2546108985375492445\_47455707435340'

app/helpers/application\_helper.rb:439:in `render\_tabs'

app/views/settings/edit.html.erb:5:in `\_app\_views\_settings\_edit\_html\_erb\_\_\_3124154614993228746\_47455728123480'

app/controllers/settings\_controller.rb:33:in `index'

lib/redmine/sudo\_mode.rb:65:in `sudo\_mode'

- **project\_list\_defaults** on /projects:

2021-02-22 09:19:53 +0100 (1) Completed 500 Internal Server Error in 65ms (ActiveRecord: 7.9ms)

2021-02-22 09:19:53 +0100 (1)

2021-02-22 09:19:53 +0100 (1) NoMethodError (undefined method `project\_list\_defaults' for #<Class:0x0000565246162c38>):

```
2021-02-22 09:19:53 +0100 (1)
2021-02-22 09:19:53 +0100 (1) app/models/project_query.rb:83:in `default_columns_names'
app/models/query.rb:731:in `columns'
app/models/query.rb:786:in `has_custom_field_column?'
app/models/project_query.rb:102:in `results_scope'
app/controllers/projects_controller.rb:278:in `project_scope'
app/controllers/projects_controller.rb:51:in `index'
lib/redmine/sudo_mode.rb:65:in `sudo_mode'
```

## Environment and upgrade

I use (and also migrate from) docker with an image based on the official docker redmine image, with additional plugins (RedmineUP-CRM/Helpdesk and some other minor plugins, see below).

In the log I could see that a few db migrations have been executed. Comparing with the redmine source these seem to be the expected migrations for the upgrade. The migrations are also logged in the database in table "schema\_migrations".

I could find no migrations in the source that would insert the added setting values (although I have seen no migrations about added settings at all - so I suspect there should be a different mechanism anyway to add new/missing settings to the database).

I tried to insert the missing settings with empty values, but I couldn't get rid of the error. Probably the values are more complex (serialized ruby objects?).

At first I had a similar problem with settings specific to the CRM plugin, but the RedmineUp support supported me with a patched version of the plugin which solved their problem. Comparing their fix they added some lazy initialization handling accessing their settings. So maybe the same mechanism would apply for the problematic settings here.

There could be more new setting values I didn't stumble upon yet. Looking at the changes there are several more additions since I installed the previous redmine version, but I have not checked if they have already been released.

Here's my configuration:

```
Environment:
  Redmine version      4.1.1.stable
  Ruby version         2.6.6-p146 (2020-03-31) [x86_64-linux]
  Rails version        5.2.4.2
  Environment          production
  Database adapter     PostgreSQL
  Mailer queue         ActiveJob::QueueAdapters::AsyncAdapter
  Mailer delivery      smtp
SCM:
  Subversion           1.10.4
  Mercurial            4.8.2
  Bazaar              2.8.0
  Git                  2.20.1
  Filesystem
Redmine plugins:
  redmine_contacts    4.3.2
  redmine_contacts_helpdesk 4.1.11
  redmine_shared_api  0.0.9
  redmine_silencer    0.4.3
```

## History

**#1 - 2021-02-22 12:41 - Michael Fuchs**

- Status changed from New to Resolved

I figured out the reason for the problem. I was using the config/settings.yml from my backup instead from using the one from the redmine release. Here are the settings defined, but not stored, that are managed in the database (I got this wrong and stored this in a volume). So my old version originally taken from the 4.0.4 release didn't contain the new definitions and default values for 4.1.1.

Sorry for bothering, you can set this ticket to invalid or better delete it... ;-)

**#2 - 2021-02-22 15:28 - Bernhard Rohloff**

- *Status changed from Resolved to Closed*

- *Resolution set to Invalid*

Michael Fuchs wrote:

...  
Sorry for bothering, you can set this ticket to invalid or better delete it... ;-)

Nothing wrong about it. :-)