redmine.org-copy202205 - Vote #70275

Email notification: bug, when number of recipients more then 8

2022/05/09 15:42 - Admin Redmine

:	Closed		:	2011/07/06	
:			:		
:			:	0%	
:	Email notifications_9		:	0.00	
:	1.3.0_36		:	0.00	
Redmine.org_URL:	https://www.redmine.org/issues/8751		**status_id:**	5	
category_id:	9		**tracker_id:**	1	
version_id:	36		**plus1:**	0	
issue_org_id:	8751		**affected_version:**		
author_id:	11698		**closed_on:**		
assigned_to_id:	0		**affected_version_id:**	35	
comments:	21				

Hello!
I updated redmine to 1.2.0 version, and now have a bug: if numder of recipients in email notifications more then 8 - everybody of them received 'bad' message: with all tags and technical information.
Parts of logs I posted on a forum, You can find it "here":http://www.redmine.org/boards/2/topics/24991

journals

I set settings: use BCC in emails. And emails start to work good! But if I don't set use BCC - bug is coming back.

I got the same problem after updating to 1.2.1. When there are 6 or 7 recipients (To and/or CC), it happened to me. Is there any limitation on http header size?

I hope this problem can be fix soon!

Thanks.

same here!

+1

The root cause is TMail version in ActionMailer.
There is a strange issue with TMail version 1.2.6 - 1.2.7.1.

(Redmine 1.1.x used Rails 2.3.5 and ActinMailer 2.3.5 used TMail 1.2.3, so it was ok.)

- Wrong adresses in headers
- https://github.com/mikel/tmail/issues/7

  I discover strange issue with TMail version 1.2.6 - 1.2.7.1.
  TMail adds unnecessary line breaks to headers with multiple email adresses.

For example this code:

m=TMail::Mail.new
m.to = ["a.user@example.com", "v.user2@example.com", "e.smith@example.com", "info@example.com", "v.pupkin@example.com"]
puts m.encoded

Produces string with empty lines (double line breaks: "\r\n\t\r\n\"):

=> "To: a.user@example.com,\r\n\t v.user2@example.com,\r\n\t e.smith@example.com,\r\n\t\r\n\t tinfo@example.com,\r\n\t v.pupkin@example.com\r\n\r\n"

This mean end of headers part of email (and broke some clients like Outlook 2003).

The fix is here: https://github.com/eac/tmail/commit/05940a3c9882147b090b061f7242a796515a2d1e.

Or downgrade to TMail 1.2.3.

Most of the changes in eac's patch are whitespace fixes. I skipped those and only made the changes in fold_the_string(), but it didn't seem to fix the problem (maybe I missed something or maybe it was based on some other revision). So I also applied the few changes to header.rb and encode.rb as noted at https://github.com/mikel/tmail/issues/7. Outlook is happy now.

Note that there are TMail packages at two locations

- REDMINE_ROOT\vendor\rails\actionmailer\lib\action_mailer\vendor\tmail-1.2.7\

*

RUBY_ROOT\lib\ruby\gems\1.8\gems\actionmailer-2.3.11\lib\action_mailer\vendor\tmail-1.2.7\

Alexander Kulemin wrote:

> I set settings: use BCC in emails. And emails start to work good! But if I don't set use BCC - bug is coming back..

I used the same workaround as Alexander to be able to upgrade my production with Redmine version:1.2.0. I'm happy to see that the problem is identified and I may integrate the patch/fix if Redmine is not fixed asap... Please target this ticket to the next bug fix release of branch 1.2.x ... Such regression is quite bad (see example below):

!redmine_email_recipients_problem.png!

Kind regards, Gilles

Gilles Cornu wrote:

> Please target this ticket to the next bug fix release of branch 1.2.x ...

Actually, it is a TMail issue, so how could we fix Redmine??

Etienne Massip wrote:

> Gilles Cornu wrote:
>
> > Please target this ticket to the next bug fix release of branch 1.2.x ...
>
> Actually, it is a TMail issue, so how could we fix Redmine??

A buggy version of TMail library (1.2.7, problem is present since 1.2.6 and still not solved in latest release, 1.2.7.1) is bundled in Rails 2.3.11, which is the [[RedmineInstall|current requirement for Redmine 1.2.x]]. Even worse, the "tmail fix": https://github.com/eac/tmail/commit/05940a3c9882147b090b061f7242a796515a2d1e has not yet been merged into https://github.com/mikel/tmail/commits/master (even no pull request opened so far).

I see several option to make Redmine release(s) *more robust for Integration Regressions* (by preference order):

include/freeze Rails framework (and maybe other libs) directly into redmine

packages/releases. This way it would be possible to (temporarily) fork/fix the library bug, and thus be more responsive for Redmine Users. Deployment would be also easier. I don't know how easy it is to do so... (I have similar experiences in PHP world, where I systematically package the complete Symfony 1.4 framework, with customization/hacks if needed)

Increase votes/pressure on Rails 2.3.x (latest was 2.3.14) and TMail 1.2.x to obtain ASAP a fixed release. Once done, change Redmine documentation...

I know it's underlined_uninteresting stuff from Redmine Developer point of view, but such kind of bugs are critical for end-user acceptance (hopefully we have here the BCC workaround) and therefore Redmine committers + community have strong interest to contribute to these dependencies...

My conclusion: don't reject the issue, but maybe change the category or tracker to something like *Integration Defect*, that clearly expresses that Redmine code is not guilty, but that Redmine Release Management cares about ASAP integrates a library that works. What's your opinion ?

Regards, Gilles

Gilles Cornu wrote:

> My conclusion: don't reject the issue, but maybe change the category or tracker to something like *Integration Defect*,

=> switch category to 'third-party libraries' ?

I guess that Web Frameworks (even more complex) meet now the same debate as the old one C/C++ debate between dynamic or static linking... see
http://en.wikipedia.org/wiki/Static_library#Advantages_of_static_linking

Gilles Cornu wrote:

> A buggy version of TMail library (1.2.7, problem is present since 1.2.6 and still not solved in latest release, 1.2.7.1) is bundled in Rails 2.3.11, which is the [[RedmineInstall|current requirement for Redmine 1.2.x]].

I will update [[RedmineInstall#Requirements|requirements]] to include a warning about TMail 1.2.7 version.

> Even worse, the "tmail fix":https://github.com/eac/tmail/commit/05940a3c9882147b090b061f7242a796515a2d1e has not yet been merged into https://github.com/mikel/tmail/commits/master (even no pull request opened so far).

I think that the problem is the lack of activity on the TMail project, probably explained by the switch by the RoR project to the use of the Mail gem (by the same author, https://github.com/mikel/mail) as an external dependency for Rails 3.

> **include/freeze Rails framework (and maybe other libs) directly into redmine packages/releases. This way it would be possible to (temporarily) fork/fix the library bug, and thus be more responsive for Redmine Users. Deployment would be also easier.**

Downloadable Redmine packages already includes Rails and its dependencies, including TMail. But there is few probabilities that the RM team will ever tweak the bundled files, sounds like a dirty thing.

> **Increase votes/pressure on Rails 2.3.x (latest was 2.3.14) and TMail 1.2.x to**

obtain ASAP a fixed release. Once done, change Redmine documentation...

Feel free to increase pressure on the TMail issue.

> I know it's <u>uninteresting stuff</u> from Redmine Developer point of view, but such kind of bugs are critical for end-user acceptance (hopefully we have here the <u>BCC workaround</u>) and therefore Redmine committers + community have strong interest to contribute to these dependencies...

There is not really more useful contribution to be done: the issue is known from the TMail author and pretty well identified.

> My conclusion: don't reject the issue, but maybe change the category or tracker to something like *Integration Defect*, that clearly expresses that Redmine code is not guilty, but that Redmine Release Management cares about ASAP integrates a library that works. What's your opinion ?

This issue will eventually be rejected since this is not a Redmine defect, I keep it open for now because it's more obvious if someone is looking for it.

As Kiwamu said, the solution is to either monkey patch TMail or downgrade it.

Etienne Massip wrote [...]

Etienne, I agree with all your arguments. IMHO it means that Rails projects (like Redmine) are forced to upgrade 'soon' to latest major version of Rails (after basic googling, it seems there is no 'long term support' goal for 2.3.x). Shouldn't Redmine team now raises the priority of #4796 ?

Upgrading to Rails 3 is not absolutely needed since Rails 2.3 series is still supported by RoR team and works pretty well (except for this regression); JPL has not stated yet about Ruby 1.9 and Rails 3.1 support (which itself requires Ruby 1.9 support).

Of course, this support will end some day, and upgrading will then become a priority, but there is some time remaining since Rails 3 still does not support plugins and database migrations (Rails 2.3 does). Rails 3.1 will, and RM will then be able to move to it.

By the way, if you look carefully at #4796, you'll see that Toshi has already made some changes in trunk and external repos towards both Ruby 1.9 and Rails 3 support.

Etienne Massip wrote:

> Gilles Cornu wrote:
>
>> Please target this ticket to the next bug fix release of branch 1.2.x ...
>
> Actually, it is a TMail issue, so how could we fix Redmine??

I've noticed that Redmine already includes some Dependency-Hotfixes by overriding external library code in "*config/initializers/10-patches.rb*":
http://www.redmine.org/projects/redmine/repository/changes/trunk/config/initializers/10-patches.rb

Actually the upgrade to Rails 2.3.11 (#6887) needed another patch on TMail 1.2.7, see r4904.

Therefore, I would propose to apply the "TMail patch from 'eac'": https://github.com/eac/tmail/commit/05940a3c9882147b090b061f7242a796515a2d1e#L0R555 directly into redmine code, until Redmine can be integrated with a fixed version of Rails 2.3.x/TMail 1.2.x

This is the monkey patch I'm using (placed in %redmine%/config/initializers)

Drew Keller wrote:

> This is the monkey patch I'm using (placed in %redmine%/config/initializers)

Awesome! it perfectly works and this is quite nicer way to quick fix a Redmine Setup. Thanks a lot Drew!

I do vote to integrate this quick-fix ASAP into branch 1.2 (until TMail is fixed or Redmine upgrades to Rails 3.1.x)

Right.

Etienne Massip wrote:

> Right.

Redmine works without 11-tmail_multiple_recipients_patch.rb.

I am getting error message as follows when I place it in initializers folder.

[git@localhost redmine-1.2.1]$ rake db:migrate:plugins RAILS_ENV=production
rake aborted!
/home/git/redmine-1.2.1/config/initializers/11-tmail_multiple_recipients_patch.rb:1: syntax error, unexpected '<'
<!DOCTYPE html PUBLIC "-//W3C//...
^
/home/git/redmine-1.2.1/config/initializers/11-tmail_multiple_recipients_patch.rb:2: syntax error, unexpected '<'

^
/home/git/redmine-1.2.1/config/initializers/11-tmail_multiple_recipients_patch.rb:2: syntax error, unexpected tIDENTIFIER, expecting $end

^

Tasks: TOP => db:migrate:plugins => environment
(See full trace by running task with --trace)

---

kiran patil wrote:

> Redmine works without 11-tmail_multiple_recipients_patch.rb.

I believe you will only see the problem on certain email clients. Notably for me, Outlook 2002 which more than half of my users have. Other email clients may be able to parse through the headers.

> I am getting error message as follows when I place it in initializers folder.
>
> [git@localhost redmine-1.2.1]$ rake db:migrate:plugins RAILS_ENV=production
> rake aborted!
> /home/git/redmine-1.2.1/config/initializers/11-tmail_multiple_recipients_patch.rb:1: syntax error, unexpected '<'
> <!DOCTYPE html PUBLIC "-//W3C//...
> ^

This looks like the start of web page code instead of ruby code. I'm guessing the file didn't download correctly.

Drew's patch attachment:11-tmail_multiple_recipients_patch.rb applied in r7863.

related_issues

relates,Closed,4796,Rails 3 support
relates,Closed,9157,Emails in Outlook doesnt work properly

#1 - 2022/05/10 17:18 - Admin Redmine

- *Email notifications_9*

- *1.3.0_36*