

Redmine - Defect #24924

Redmine installation produces different results depending on the date installed

2017-01-25 21:59 - Angelo Bertoli

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Gems support	Estimated time:	0.00 hour
Target version:		Affected version:	3.3.1
Resolution:	Wont fix		

Description

The general problem

Redmine releases leave their dependencies open-ended, and assume that later versions of gems will always be backward compatible and can be installed with the release.

If you install Redmine on a system using the instructions on the site, you will get a different result depending on what version of a gem happens to be available at that time.

A side effect, specific example of this being an issue

When the Gemfile specifies a version like '>= x.x.x' it will throw an error during installation if the newest version of a gem has changed dependencies.

e.g.

- When Redmine 3.3.1 was released it could be installed on RHEL 7 using Ruby 2.0
- At some point in time, nokogiri 1.7 was released that requires Ruby 2.1
- When attempting to run bundler to install Redmine on an identical system after this point will generate an error requiring Ruby 2.1

Solution

The current/correct practice for maintaining a Gemfile should be to use '~> x.x.x' whenever possible. This ensures both that (1) the version of the gem that was verified with a particular release is used, and (2) future gem versions do not cause unresolved dependency problems.

Associated revisions

Revision 16264 - 2017-01-27 05:21 - Toshi MARUYAMA

Gemfile: remove builder (#24924)

Rails 4.2.7.1 has 'builder', '~> 3.1'.

<https://github.com/rails/rails/blob/v4.2.7.1/actionview/actionview.gemspec#L24>

History

#1 - 2017-01-25 22:00 - Angelo Bertoli

This issue affects more than one version, but should be fixed only for future versions.

#2 - 2017-01-26 03:02 - Toshi MARUYAMA

- Status changed from New to Closed

- Resolution set to Wont fix

Angelo Bertoli wrote:

- At some point in time, nokogiri 1.7 was released that requires Ruby 2.1

[r16169](#) fixed it.

And we have CI server.

<http://www.redmine.org/builds/index.html>

Gemfile should be to use '~> x.x.x'

It is too hard to follow new version release.

#3 - 2017-01-26 04:06 - Angelo Bertolli

I see the example of nokogiri is changed.

Friendly suggestion:

You can use your CI server for normal development, but when you're ready to make a release, your workflow could be to update the Gemfile for that release to specify the snapshot of the condition that Redmine was tested for the release at that time.

The problem of gems updating is well-known, and if configuration management is not handled properly, you will continue to have endless questions (not only on this site but elsewhere) for help when the installation doesn't work out of the box.

Just something to consider.

#4 - 2017-01-26 06:17 - Toshi MARUYAMA

What gem do you suggest use '~> x.x.x'?

Our gem policy is using fixed Rails version because Rails minor release broke Redmine many times.

The reason of 'gem "nokogiri", ">= 1.6.7.2"' is security advisories ([r15096](#)).

#5 - 2017-01-26 16:57 - Angelo Bertolli

You are right, this isn't really a bug that will correspond to a code fix, but rather a possible process improvement.

My thought is whatever is in trunk/branch can stay with '>=' but when you produce a release, e.g. tag and zip: that one can be converted to '~>' using the version that works with your CI server, because it is the configuration that was tested at the time of release.

I see things like builder, rails-html-sanitizer, rmagick, etc. in trunk that may break future deployments of a given version of Redmine.

#6 - 2017-01-27 05:09 - Toshi MARUYAMA

Angelo Bertolli wrote:

I see things like builder, rails-html-sanitizer, rmagick, etc. in trunk that may break future deployments of a given version of Redmine.

builder and rails-html-sanitizer dependencies are defined by Rails.

<https://github.com/rails/rails/blob/v4.2.7.1/actionview/actionview.gemspec#L24>

#7 - 2017-01-27 05:57 - Toshi MARUYAMA

I have sent pull request of "rails-html-sanitizer >= 1.0.3".

<https://github.com/rails/rails/pull/27814/>

#8 - 2017-01-27 06:45 - Angelo Bertolli

I apologize--I'm not knowledgeable enough about Ruby development to fully understand the implications of Rails dependencies with regards to gems and Redmine. Thanks for taking the time to look into this.