

Foreman - Bug #12219

Test failures on Ruby 2.2

10/20/2015 04:44 AM - Dominic Cleal

Status:	Closed	
Priority:	Normal	
Assignee:	Eric Helms	
Category:	Tests	
Target version:	1.11.0	
Difficulty:		Fixed in Releases:
Triaged:		Found in Releases:
Bugzilla link:		Red Hat JIRA:
Pull request:	https://github.com/theforeman/foreman/pull/2990	

Description

Ruby 2.2 support in [#11016](#) is incomplete, it still has test failures:

On PostgreSQL it's failing to initialise the pg gem:

```
08:28:36 + bundle exec rake db:drop db:create db:migrate
08:28:47 Couldn't drop test_develop_pr_core_ruby22-1-dev : #<LoadError: Please install the postgresql adapter: `gem install activerecord-postgresql-adapter` (incompatible library version - /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/pg-0.18.3/lib/pg_ext.so)>
08:28:47 Couldn't drop test_develop_pr_core_ruby22-1-test : #<LoadError: Please install the postgresql adapter: `gem install activerecord-postgresql-adapter` (incompatible library version - /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/pg-0.18.3/lib/pg_ext.so)>
08:28:47 rake aborted!
08:28:47 LoadError: Please install the postgresql adapter: `gem install activerecord-postgresql-adapter` (incompatible library version - /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/pg-0.18.3/lib/pg_ext.so)
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/lib/active_support/dependencies.rb:251:in `require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/lib/active_support/dependencies.rb:251:in `block in require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/lib/active_support/dependencies.rb:236:in `load_dependency'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/lib/active_support/dependencies.rb:251:in `require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/pg-0.18.3/lib/pg.rb:4:in `<top (required)>'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/lib/active_support/dependencies.rb:251:in `require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/lib/active_support/dependencies.rb:251:in `block in require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/lib/active_support/dependencies.rb:236:in `load_dependency'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/lib/active_support/dependencies.rb:251:in `require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib/active_record/connection_adapters/postgresql_adapter.rb:8:in `<top (required)>'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/lib/active_support/dependencies.rb:251:in `require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/lib/active_support/dependencies.rb:251:in `block in require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/lib/active_support/dependencies.rb:236:in `load_dependency'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/lib/active_support/dependencies.rb:251:in `require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib/active_record/connection_adapters/abstract/connection_specification.rb:50:in `resolve_hash_connec
```

```
tion'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib
/active_record/connection_adapters/abstract/connection_specification.rb:29:in `spec'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib
/active_record/connection_adapters/abstract/connection_specification.rb:130:in `establish_connecti
on'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib
/active_record/railties/databases.rake:101:in `create_database'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib
/active_record/railties/databases.rake:62:in `block (3 levels) in <top (required)>'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib
/active_record/railties/databases.rake:62:in `each'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib
/active_record/railties/databases.rake:62:in `block (2 levels) in <top (required)>'
08:28:47 LoadError: incompatible library version - /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_
core_ruby22-1/gems/pg-0.18.3/lib/pg_ext.so
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/li
b/active_support/dependencies.rb:251:in `require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/li
b/active_support/dependencies.rb:251:in `block in require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/li
b/active_support/dependencies.rb:236:in `load_dependency'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/li
b/active_support/dependencies.rb:251:in `require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/pg-0.18.3/lib/pg.rb:4:i
n `<top (required)>'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/li
b/active_support/dependencies.rb:251:in `require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/li
b/active_support/dependencies.rb:251:in `block in require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/li
b/active_support/dependencies.rb:236:in `load_dependency'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/li
b/active_support/dependencies.rb:251:in `require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib
/active_record/connection_adapters/postgresql_adapter.rb:8:in `<top (required)>'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/li
b/active_support/dependencies.rb:251:in `require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/li
b/active_support/dependencies.rb:251:in `block in require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/li
b/active_support/dependencies.rb:236:in `load_dependency'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activesupport-3.2.22/li
b/active_support/dependencies.rb:251:in `require'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib
/active_record/connection_adapters/abstract/connection_specification.rb:50:in `resolve_hash_connec
tion'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib
/active_record/connection_adapters/abstract/connection_specification.rb:29:in `spec'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib
/active_record/connection_adapters/abstract/connection_specification.rb:130:in `establish_connecti
on'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib
/active_record/railties/databases.rake:101:in `create_database'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib
/active_record/railties/databases.rake:62:in `block (3 levels) in <top (required)>'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib
/active_record/railties/databases.rake:62:in `each'
08:28:47 /usr/local/rvm/gems/ruby-2.2.3@test_develop_pr_core_ruby22-1/gems/activerecord-3.2.22/lib
/active_record/railties/databases.rake:62:in `block (2 levels) in <top (required)>'
08:28:47 Tasks: TOP => db:create
08:28:47 (See full trace by running task with --trace)
```

http://ci.theforeman.org/job/test_develop_pr_core_ruby22/7/database=postgresql.ruby=2.2/console

On sqlite3 and mysql, it actually runs the test suite and fails with:

```

08:29:55 =====
08:29:55 Failure:
08:29:55 <"localhost"> expected but was
08:29:55 <"foreman.some.host.fqdn">.
08:29:55 test_0001_foreman server header is set(ApplicationMailerTest)
08:29:55 test/unit/application_mailer_test.rb:14:in `block in <class:ApplicationMailerTest>'
08:29:55 =====

08:30:23 =====
08:30:23 Error: test_0014_add compute resource(ComputeResourceTest): NameError: uninitialized clas
s variable @@current in MiniTest::Unit::TestCase
08:30:23 test/unit/compute_resource_test.rb:109:in `block in <class:ComputeResourceTest>'
08:30:23 =====

08:33:33 =====
08:33:33 Error: test_0001_be tested(MenuItemTest::MenuItem#caption): NoMethodError: undefined meth
od `skip' for #<#<Class:0x0000000c8fccb0>:0x0000000ad272d8>
08:35:30 =====
08:37:17 E
08:37:17 =====

```

(http://ci.theforeman.org/job/test_develop_pr_core_ruby22/7/database=sqlite3.ruby=2.2/console,
http://ci.theforeman.org/job/test_develop_pr_core_ruby22/7/database=mysql.ruby=2.2/console)

The mysql test failed with a SIGKILL due to the OOM killer, this may require some consideration given to our Jenkins infrastructure. The RSS of the Ruby process was reported to be 2.54GB.

Related issues:

Related to Foreman - Bug #11016: missing gem in test.rb under ruby 2.2.0	Closed	07/05/2015
Related to Foreman - Refactor #15753: Unpin pg gem	Closed	07/20/2016
Blocks Packaging - Feature #7228: Rebuild packages under ror41/ruby22 SCLs	Closed	08/22/2014
Blocked by Foreman - Feature #7230: Upgrade Ruby on Rails to 4.1	Closed	08/22/2014

Associated revisions

Revision 7d54aa32 - 12/21/2015 09:31 AM - Eric Helms

Fixes #12219: Pin pg to 0.15.X

The pg gem from 0.16.0 to 0.18.4 breaks the ability to run database rake commands due to an improper linking when ruby-devel is installed and trying to use Ruby greater than or 2.2.2

Revision 31999642 - 12/21/2015 10:09 AM - Dominic Cleal

refs #12219 - pin pg to 0.15.x

Revision 859e1e74 - 01/04/2016 09:03 AM - Dominic Cleal

refs #12219 - relax pg pin for Fedora 21's pg 0.17.1

The package in Fedora wouldn't be susceptible to the multi-lib loading issue that prompted ticket #12219, so ignore the upper pin.

History

#1 - 10/20/2015 04:44 AM - Dominic Cleal

- Related to Bug #11016: missing gem in test.rb under ruby 2.2.0 added

#2 - 11/09/2015 05:41 AM - Dominic Cleal

- Blocks Feature #7228: Rebuild packages under ror41/ruby22 SCLs added

#3 - 11/11/2015 07:19 AM - Anonymous

- Status changed from New to Assigned

- Assignee set to Anonymous

#4 - 12/14/2015 03:12 PM - Eric Helms

During some local testing I noticed that the postgres related failure only occurred with 2.2.3. Using 2.2.1 (the default I got when I ran 'rvm install 2.2') and 2.2.2 I had no issues. Given that the SCL's current versions are 2.2.0 and 2.2.2, would the fact it runs fine on 2.2.1 and 2.2.2 be enough to turn on 2.2 support? (Assuming of course the sqlite and mysql issues were fixed).

#5 - 12/15/2015 02:47 AM - Dominic Cleal

We should use the latest 2.2.x I think, not just what happens to be in the SCL. Other platforms (such as Debian 9) may use 2.2.3, as might developers.

The test failures seen on sqlite and MySQL seem to be fixed on the Rails 4 branch under ticket [#7230](#).

#6 - 12/15/2015 08:35 PM - Eric Helms

Some updated findings, locally I have found that the 0.15.1 version of the pg gem does not lead to this error, but every version above that does. Currently, we package and distribute 0.12.2 version of the pg gem via RPM and the bundler asset is set to '~> 0.11'. One solution could be to set ourselves to the 0.15.1 version of the pg gem and package that. I have not been able to determine from diffing why 0.16.0 breaks but 0.15.1 works.

#7 - 12/16/2015 02:53 AM - Dominic Cleal

Note that the version in the bundler.d file is designed to [match the AR adapter](#), which in 4.1 is ~> 0.11 still. 0.15 is required for Ruby 2.2 compatibility ([bug](#)) while Rails master has moved up to ~> 0.18. 4.1 ought to work fine with 0.18 still.

What's the underlying error? Presumably it's just catching a LoadError and reporting a generic message.

#8 - 12/16/2015 11:55 AM - Eric Helms

Some further information into this, based on Daniel's inability to re-produce on a new setup I spun up new versions of centos6-devel and centos7-devel from katello-deploy. I then installed 2.2.3 and 2.2.4 (released Dec 16th) via rvm install, bundle installed and ran the command. There was no error in any of those cases. The only time I am able to reproduce this is from an existing, older rvm setup on a devel box of mine. The version of RVM is the same so this may be related to how RVM is built/installed and newer libraries? I have zero leads as to the root cause but figured I'd share that bit of information.

#9 - 12/18/2015 10:08 AM - Eric Helms

Ran the tests on sqlite3 + 2.2.4 with latest (i.e. post Rails 4 merge) and everything passed for me. Which would in theory leave the lingering postgresql issue that seems to be present only on older RVM installations which would largely affect Jenkins. I am still working to see if I can "fix" an existing RVM installation.

#10 - 12/18/2015 10:17 AM - Dominic Cleal

I managed to reproduce myself on one of the slaves that reported it, looking into it more now. I get the same error that we saw on Rails 3.2 when requiring the gem directly, so the different error on Rails is probably just a rescue.

```
[dcleal@slave02 foreman]$ rails console
/usr/local/rvm/gems/ruby-2.2.3@dcleal/gems/activesupport-4.1.5/lib/active_support/dependencies.rb:247:in `require': Could not load 'active_record/connection_adapters/postgresql_adapter'. Make sure that the adapter in config/database.yml is valid. If you use an adapter other than 'mysql', 'mysql2', 'postgresql' or 'sqlite3' add the necessary adapter gem to the Gemfile. (LoadError)
)
   from /usr/local/rvm/gems/ruby-2.2.3@dcleal/gems/activesupport-4.1.5/lib/active_support/dependencies.rb:247:in `block in require'
   from /usr/local/rvm/gems/ruby-2.2.3@dcleal/gems/activesupport-4.1.5/lib/active_support/dependencies.rb:232:in `load_dependency'
   from /usr/local/rvm/gems/ruby-2.2.3@dcleal/gems/activesupport-4.1.5/lib/active_support/dependencies.rb:247:in `require'
   from /usr/local/rvm/gems/ruby-2.2.3@dcleal/gems/pg-0.18.4/lib/pg.rb:4:in `'
[dcleal@slave02 foreman]$ bundle exec irb
2.2.3 :001 > require 'pg'
LoadError: incompatible library version - /usr/local/rvm/gems/ruby-2.2.3@dcleal/gems/pg-0.18.4/lib/pg_ext.so
   from /usr/local/rvm/gems/ruby-2.2.3@dcleal/gems/pg-0.18.4/lib/pg.rb:4:in `require'
   from /usr/local/rvm/gems/ruby-2.2.3@dcleal/gems/pg-0.18.4/lib/pg.rb:4:in `'
   from (irb):1:in `require'
   from (irb):1
   from /usr/local/rvm/rubies/ruby-2.2.3/bin/irb:11:in `'
```

#11 - 12/18/2015 10:45 AM - Dominic Cleal

The pg_ext.so library is being linked to the wrong Ruby, even when using rvm. It's being linked to the system Ruby, which isn't going to be ABI compatible with rvm's.

Here's ldd from a git checkout of the pg gem, compiled under rvm's 2.2.3 but is linked to EL6's Ruby 1.8:

```
[dcleal@slave02 ruby-pg]$ ldd ./tmp/x86_64-linux/pg_ext/2.2.3/pg_ext.so | grep ruby
libruby.so.1.8 => /usr/lib64/libruby.so.1.8 (0x00007fc20c1ab000)
```

A bisect shows this is ever since <https://github.com/ged/ruby-pg/commit/a6ef4bd3a5ba8dce79331f0678dee315813ac754>.

rvm on our slaves is currently version 1.26.11. I don't see this on my local Fedora system under rvm 1.25.15.

It appears that the tfm-rubygem-pg RPM I built for EL6 is linked against libruby2.2, so it may be a combination of rvm and pg.

#12 - 12/18/2015 12:18 PM - Dominic Cleal

Something, probably in mkmf, changed in Ruby 2.2 to cause lib paths (used for linking) to be **prepended** instead of appended when building native extensions. Comparing 2.0 and 2.2, I see the rpath of the generated pg_ext.so is:

- /usr/local/rvm/rubies/ruby-2.0.0-p643/lib:/usr/lib64
- /usr/lib64:/usr/local/rvm/rubies/ruby-2.2.3/lib

The /usr/lib64 is added because it's the path of the PostgreSQL libraries. When it's prepended though, any libruby in /usr/lib64 is linked instead of the RVM version - this causes the linking against the wrong Ruby version and later, a runtime failure.

This happens on our slaves but not on a clean EL6 installation because we have ruby-devel installed (this package might need to be installed to reproduce the bug).

I'd suggest for now that we pin the pg gem to < 0.16 and then this needs either changing in pg (which moved to using mkmf instead of appending manually to the link path in the breaking commit) or mkmf/Ruby core.

#13 - 12/18/2015 12:45 PM - Eric Helms

+1 to your assessment - uninstalling ruby-devel on my "broken" box and installing pg afterward provided the correct linking.

#14 - 12/18/2015 12:50 PM - The Foreman Bot

- Status changed from Assigned to Ready For Testing
- Pull request <https://github.com/foreman/foreman/pull/2990> added

#15 - 12/21/2015 09:40 AM - Dominic Cleal

- Assignee changed from Anonymous to Eric Helms

#16 - 12/21/2015 10:02 AM - Eric Helms

- Status changed from Ready For Testing to Closed
- % Done changed from 0 to 100

Applied in changeset [7d54aa3288e394302b8d175ed2c2fbfa5e4a2e67](https://github.com/foreman/foreman/commit/7d54aa3288e394302b8d175ed2c2fbfa5e4a2e67).

#17 - 12/22/2015 06:15 AM - Dominic Cleal

- Blocked by Feature #7230: Upgrade Ruby on Rails to 4.1 added

#18 - 07/20/2016 08:48 AM - Dominic Cleal

- Related to Refactor #15753: Unpin pg gem added