Ruby - Bug #13040

syslog/logger uses "require 'logger'" which is interpreted as circular dependency

12/15/2016 07:11 PM - lemsx1 (Luis Mondesi)

Status:	Rejected		
Priority:	Normal		
Assignee:			
Target version:			
ruby -v:	ruby 2.3.1p112 (2016-04-26 revision 54768) [x86_64-linux]	**Backport:**	2.1: UNKNOWN, 2.2: UNKNOWN, 2.3: UNKNOWN

Description

while using the 'syslog/logger' require from Ruby it works well, however if one tries to use it via "test" it complaints about circular require and the constant "Logger" not defined. After a little digging, I realized that 'syslog/logger' has a line that says require 'logger' which could mean itself!

The fix is to use require_relative '../logger' as this is what it intends in order to piggy back on Logger:

Current code for syslog/logger:

1. # frozen_string_literal: false

2. require 'syslog'
3. require 'logger'

Fixed code:

1. # frozen_string_literal: false

2. require 'syslog'
3. require_relative '../logger'

History

#1 - 12/16/2016 01:15 AM - nobu (Nobuyoshi Nakada)

'syslog/logger.rb' and 'logger.rb' are different files.
What do you mean by "test"?

#2 - 12/16/2016 01:15 AM - nobu (Nobuyoshi Nakada)

- Status changed from Open to Feedback

#3 - 12/20/2016 02:33 AM - lemsx1 (Luis Mondesi)

Nobuyoshi Nakada wrote:

> 'syslog/logger.rb' and 'logger.rb' are different files.
> What do you mean by "test"?

For example, I setup a new Rails 5.0.0.1 app and minitest for use with "spec" via a test_helper file, nothing out of the norm. Then ran "rake test" and saw a complaint (warning) about "circular dependency". If I turn the warnings off, then you won't see the message, but would fail saying logger constant is not defined. It took me a little while to realize that both syslog/logger and "logger" are both in the path while "require 'logger'" is called from syslog/logger.

And like I said above, require_relative disambiguate the call and makes it very clear that you mean to require the "logger" from the previous directory.

While my app runs I do not get this warning or error, it only happens in "test mode". Whatever minitest does to run tests causes Ruby to get confused.

#4 - 05/18/2017 01:55 PM - lemsx1 (Luis Mondesi)

Am I the only person having this problem? This is in different code sources we are maintaining. Here is the latest. In a new project I setup test with minitest/spec

1. added pp $LOAD_PATH to my test_helper.rb

2. rake test

$> rake test
["/Users/lmondesi/Projects/new_project/lib",
"/Users/lmondesi/Projects/new_project/lib/new_project",
"/usr/local/lib/ruby/gems/2.4.0/gems/rake-12.0.0/lib",
"/usr/local/lib/ruby/gems/2.4.0/gems/did_you_mean-1.1.0/lib",
"/usr/local/lib/ruby/gems/2.4.0/gems/minitest-5.10.1/lib",
"/usr/local/lib/ruby/site_ruby/2.4.0",
"/usr/local/lib/ruby/site_ruby/2.4.0/x86_64-darwin16",
"/usr/local/lib/ruby/site_ruby",
"/usr/local/lib/ruby/vendor_ruby/2.4.0",
"/usr/local/lib/ruby/vendor_ruby/2.4.0/x86_64-darwin16",
"/usr/local/lib/ruby/vendor_ruby",
"/usr/local/Cellar/ruby/2.4.1_1/lib/ruby/2.4.0",
"/usr/local/Cellar/ruby/2.4.1_1/lib/ruby/2.4.0/x86_64-darwin16"]
/usr/local/Cellar/ruby/2.4.1_1/lib/ruby/2.4.0/syslog/logger.rb:81:in <class:Logger>': uninitialized constant Logger (NameError) from
/usr/local/Cellar/ruby/2.4.1_1/lib/ruby/2.4.0/syslog/logger.rb:48:in <top (required)>'
from /usr/local/Cellar/ruby/2.4.1_1/lib/ruby/2.4.0/rubygems/core_ext/kernel_require.rb:55:in require' from
/usr/local/Cellar/ruby/2.4.1_1/lib/ruby/2.4.0/rubygems/core_ext/kernel_require.rb:55:in require'

#5 - 06/08/2017 03:46 PM - lemsx1 (Luis Mondesi)

Ok, I was able to reproduce this now.

1. create a new Rails Project
2. create an initializer under configs/initializer named "delayed_job.rb" with:

require 'syslog/logger'

Delayed::Worker.delay_jobs = !Rails.env.test?
Delayed::Worker.logger = Rails.logger

hack to reopen syslog in local1

Syslog::Logger.syslog.close
Syslog::Logger.syslog = Syslog.open('bmanage', Syslog::LOG_ODELAY, Syslog::LOG_LOCAL1)

1. run bin/rake test or bundle exec rake test

Essentially, this was required with older libraries of Syslog::Logger. Now that Syslog::Logger allows one to change facility when initializing an instance, there is no need for this code.

After removing this code, the bug went away!

You can now close this issue.

#6 - 06/09/2017 03:21 AM - hsbt (Hiroshi SHIBATA)

- Status changed from Feedback to Rejected