« previous 1 2 3

Cross-Site-Scripting Protection By Default

In past versions of Rails, you had to manually implement content escaping when creating your views. This was usually done with the #h method, as in `<%= h(@post.body) %>` (a "blacklist" approach – "this stuff is bad"). In Rails 3, the default behaviour is to automatically escape all output, unless explicitly marked with #html_safe (a whitelist approach – "everything's bad unless I say otherwise"). For example:

Rails 2.3 - 3.0


```
<!-- Rails 2.3 -->
<%= h(@post.body) %>
<!--
escaped and safe:

&lt;script type=&quot;text/javascript&quot;&gt;
     alert('lol, u got hacked');
&lt;/script&gt;

-->
<%= @post.body %>
<!--
result:

<script type="text/javascript">
  alert('lol, u got hacked');
</script>

 * not escaped, vulnerable to cross-site-scripting attack!
-->

<!-- Rails 3 -->
<%= @post.body %>
<!--
escaped and safe

&lt;script type=&quot;text/javascript&quot;&gt;
    alert('lol, u got hacked');
 &lt;/script&gt;

-->

<%= @post.body.html_safe %>
<!--
result:

<script type="text/javascript">
  alert('lol, u got hacked');
</script>

 * not escaped, vulnerable to cross-site-scripting attack!
-->
```

Marking strings as #html_safe should be done sparingly, only in situations where it is absolutely needed, and has no possibility of having tainted input. In modern web applications such situations are fairly rare, though still possible.

Other Notable Changes

* No more script/* - just use `rails [command] [arguments]` now. For example, `rails server` will start the development server, `rails new /path/to/my/app` creates a new Rails project from the default template, and `rails generate model my_model` would create a new model.

* Named Scopes have been changed so that the keyword is now simply `scope`, **not** `named_scope`. The syntax has changed as well:

```
class Article < ActiveRecord::Base
  scope :unpublished, where(:published => false)
end
```

H OPEN HEADLINES

The H is closing down

Hardware Hacks: Fire, alarms, touchable boards and NFC rings

GitHub gets smart over open source licences

NSS 3.15.1 brings TLS 1.2 support to Firefox

Second Android signature attack disclosed

One month left for the EclipseCon Europe 2013 call for papers

H OPEN

Kernel Log: Coming in 3.10 (Part 4) - Drivers

Linux 3.10 will be able to use the video acceleration features offered by Radeon graphics cores. Systems with Intel graphics will wake from standby faster. Linux now has an input device driver for Apple's infrared receiver **more »**

The trouble with "Business Source"

The problem of creating funding in a new software business is a major one, and doubly so for open source based companies. Michael Widenius recently described his solution to the problem, "Business Source", claiming it delivers "most of the benefits of open source". The H took a look to see how that held up **more »**

Kernel Log: Coming in 3.10 (Part 3) - Infrastructure

Kernel developers have toned down an over-eager feature for protecting against the Samsung UEFI bug and added a function for reducing timer interrupt overhead. Improvements have also been made to Hyper-V support and instructions for reporting errors **more »**

Whatever happened to Google?

Bundler is now used for gem (dependency) management.

Bundler is a tool for managing gem dependencies in your Rails (or other Ruby) project, and is now the default method for managing gem dependencies in a Rails 3.x application. Using Bundler requires that your Rails application has a file named 'Gemfile' (no extension) at the directory root of your application. Though it has no extension, code inside Gemfile is evaluated as Ruby code, and is executed by the "bundle" executable that's made available when you install Bundler.

Setting up a Rails application to use Bundler is pretty straightforward. Start by installing Bundler as a gem:

```
$ gem install bundler
```

Tip: Rubygems will usually install RI and RDoc documentation by default. If you want to disable this behaviour, as it usually takes a while to generate all that documentation, create a .gemrc file in your home directory, and add the following code:

```
gem: --no-ri --no-rdoc
```

Alternatively, you can specify `'--no-ri --no-rdoc'` each time you issue the `gem install ...` command.

Once you have Bundler installed, either open the existing Gemfile in your Rails application, or if you don't have one, issue the following command:

```
$ bundle init
```

This will create a Gemfile with some sample code that looks like this:

```
# A sample Gemfile
source "http://rubygems.org"


# gem "rails"
```

As you can see, the syntax is just like that of Ruby, and code inside a Gemfile will be executed as Ruby code. (Just in case you were thinking about putting other application start-up code here, don't. That stuff belongs in an initializer instead.)

Next, you can add all your gem dependencies to the Gemfile. An example Gemfile may look something like this:

```
gem 'rails', '3.0.10' # Specify exact Rails version to use
gem 'pg' # Use latest available PostgreSQL gem

group :development, :test do
  gem "rspec-rails'"
  gem "nifty-generators"
  gem "faker"
  # ... and so on, for your dev/test gems
end

platforms :mri_19 do
  # Use the ruby 1.9 debugger
  gem "ruby-debug19",
      :require => 'ruby-debug' if RUBY_VERSION < "1.9.3"
end
```

This Gemfile example shows many useful features of Bundler:

Specifying a gem version can be done in a one-line statement, as with the first line in the above example. This tells Bundler specifically to use Rails version 3.0.9. When the Rails development team releases a new version, you can simply change the version line and run `bundle update` (more on this below). Alternatively, you can use the latest stable version of a gem by not specifying any version information. That could be a bad idea, however; Rubyists are well-known for not being all that concerned with backwards compatibility, and "upgrades" will often break things that worked before. This is why it's a good idea to lock your application down to specific gem versions.

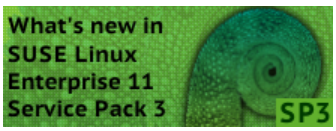
Groups are one of the best features of Bundler. Using a group block as above, you can specify – in a Rails application – gems to be loaded *only under those specific environments*. In this example, we're telling Bundler to load the listed gems only when the application is running in development or test mode. In the past, this may have been done by adding a "config.gem" line to the appropriate environment file (e.g. config/environments/test.rb). Now, we can simplify the process by including it in a group block. This is useful from a performance perspective, as you will never need to load development or test gems in a production environment, so confining those gems to the environments in which they're specifically needed avoids the performance hit on memory and startup time when the application is deployed in a production environment.

Platforms are another very useful feature. Multiple Ruby platforms exist in today's modern open source environment, including JRuby (built on the Java Virtual Machine),

Although Google continues to support a variety of open projects and people, Glyn Moody notes that, following recent changes to Google Code and Google Talk, concern is growing that something fundamental has changed **more »**

HOT ON THE H

What's new in SUSE Linux Enterprise 11 SP3



Service Pack 3 includes numerous enhancements for virtualisation and, by adding Secure Boot support and new drivers, beefs up support for newer hardware. There are also numerous enhancements relating to server storage and networking **more »**

What's new in Fedora 19

In a nod to fans of classic desktop interfaces, the new Fedora includes a MATE variant and classic mode for GNOME. Systemd now takes care of containers and assigning network names. New drivers support 3D acceleration in newer Radeon graphics cards **more »**

What's new in Linux 3.10

A second SSD caching framework and support for the new Radeons' video decoder are two of the most important enhancements in Linux 3.10, which is now out. This version also includes several new and improved drivers and a change to the network stack to speed up HTTP connections **more »**

Free Software post-PRISM

The news has been full of talk of spying, whistleblowing and data mining. Glyn Moody looks at how open source has been used to threaten freedom and privacy and how it could be used to

Rubinius, and MRI ("Matz Ruby Interpreter") versions 1.8.7 and 1.9.2, among others. Using the `platforms` feature as in the above example, you can specify which gems are loaded using which platform. This is useful if, for example, you build an application using MRI on your local machine, but want to deploy using JRuby. In this hypothetical situation, you would use a different database adapter than with MRI, and would want your application to be able to install the right gem for its particular Ruby implementation.

Finally, when you need to update your bundle (of gems), the process is simple: first, edit the Gemfile to reflect the new version information for the gems you want to update, then issue the following command:

```
$ bundle update gemname
```

Specifying the name of the gem to update tells Bundler to update only that gem to the information listed in Gemfile, but not to touch the other gem versions. If you want to update all gems, whether their version has changed or not, simply issue "bundle update" without any further arguments.

More information on Bundler can be found at the project's official web site, gembundler.com.

For More Information

More information about the changes in Rails 3 can be found in a variety of places, some of which are linked below for your convenience:

- Railscast on Bundler – Ryan Bates' screencast on Bundler in Rails 3
- Rails 3.0: It's ready! – weblog.rubyonrails.org
- Dive into Rails 3: ARel video – a video explanation of the new ActiveRecord query engine/syntax
- Rails 3 Routing Guide – guides.rubyonrails.org
- Action Mailer guide – guides.rubyonrails.org
- Railscast on XSS – Ryan Bates' screencast on XSS protection in Rails 3

This completes our overview of the changes introduced into version 3.0 of Rails since 2.3. In the second of these two articles we will look at the new features and improvements due in version 3.1, expected on 30 August.

J. Austin Hughey (@jaustinhughey) is a web applications engineer from Texas (United States). After building web applications in PHP/MySQL for six years, in 2007 he made the switch to Ruby and hasn't really looked back.

« previous 1 2 3

Print Version | Permalink: http://h-online.com/-1285884

Also on The H:

RubyGems 1.5.0 now supports Ruby 1.9.2
JRuby 1.6.0 nears with first release candidate
Rails 3 nears with release candidate availability
JRuby boosted to 1.5.0
Ruby On Rails Security Guide published as free ebook
Ruby on Rails 2.2 to be thread safe

Continuous database migration with Liquibase and Flyway

An application's version-controlled source code is stored in the repository. Why not that of the database? To reproduce arbitrary database states in development, test or production environments, two powerful Java libraries are at hand that can be seamlessly integrated into a build for an agile Continuous Delivery **more »**

Unit testing with Node.js

Consistent unit testing is a basic quality requirement in modern software rdevelopment. Mocha is a framework for writing and executing such tests in Node.js **more »**

Ruby 2.0 - the 20th birthday present

On 24 February 2013, the Ruby community celebrated the 20th birthday of its programming language. Ruby 2.0, a new major release that includes various exciting new features, was released at the same time and The H looks at some of the major changes **more »**

HITS OF THE H

Linux Mint 15: A better Ubuntu for the desktop

The Linux Mint project has announced "the most ambitious release since the start of the project". Linux Mint 15 promises a focus on the desktop that Ubuntu has been neglecting lately. The H investigates whether the release delivers on these ambitions **more »**

What's new in Linux 3.9

The Linux kernel is finally able to use SSDs as hard-disk cache. Changes to the network subsystem promise to improve the way server jobs are distributed across multiple processor cores. Linux 3.9 also includes drivers for new AMD graphics chips and soon-expected Wi-Fi components from Intel **more »**

Replacing Google Reader

For a large number of internet users the current challenge is finding a replacement for Google Reader. The H's Fabian Scherschel has looked at the functionality that made Google Reader popular and the current best alternatives to the Reader experience **more »**

Attacking TrueCrypt

TrueCrypt is considered the software of choice for encrypting data. A small utility called TCHead systematically takes on this encryption **more »**