RELIABLY DEPLOYING RAILS APPLICATIONS

Reliable, repeatable deployment & provisioning

BY BEN DIXON

Reliably Deploying Rails Applications

Reliable, repeatable deployment and provisioning

Ben Dixon

This book is for sale at http://leanpub.com/deploying_rails_applications

This version was published on 2013-10-25

Contents

1.0 - Intro

The purpose of this book

This book will show you from start to finish how to:

- Setup a VPS from Scratch
- Setup additional servers in minutes
- Use Capistrano to deploy reliably
- Automate boring maintenance tasks

If you've got applications on Heroku which are costing you a fortune, this will provide you with the tools you need to move them onto a VPS.

If you're already running your app on a VPS but the deploy process is flaky - it sometimes doesn't restart or loads the wrong version of the code - this book provides a template for making the process robust.

I've spent hundreds of hours combing through blog posts, documentation and tweaking config files. This has got me to the stage where deploying to a VPS is as easy as - in fact often easier than - deploying to Heroku. If you want to do the same, this book will save you a lot of time.

About Me

Currently based in Google Campus, London, I've been developing web applications for over eight years. Over the last few years specialising in Ruby on Rails development and deployment. I work both as a consultant - primarily to startups - and on projects of my own.

I'm the technical lead at a health and fitness startup who provide the timetabling for many of the UK's public leisure operators as well as producing an international iOS app (Speedo Fit) for swimmers sponsored by a major global fitness brand.

As part of developing the infrastucture for this I've dealt with everything from the usual rapid growth from 10's of requests per minute to 10's per second to more unusual challenges such as expanding infrastructure into China and dealing with the wrath of timezone switching.

Intended Audience

This book is intended for people who develop Ruby on Rails applications and have to be involved with or, completely manage, the infrastructure and deployment of these applications.

Whether deploying applications is new to you or you've been doing it by hand for a while and want to move to a more structured approach, I hope this book will be useful to you.

Pre-requisites

It's assumed anyone reading this is already profficient in Ruby and Rails development.

Some basic knowledge of the unix command line is assumed, in particular that you can:

- Use SSH to connect to remote servers
- Navigate around the file system from the shell (cd, ls, dir, mv, rm etc)
- Have a basic understanding of web architecture. E.g. what a server is, how to setup DNS etc.

It will be useful but not essential if you already have some familiarity with web server structure, e.g. the place of virtual host files, permissions and the like.

How to read this book

This book is structured around sample code, all of which is available on Github. In general each chapter will examine a section of the sample code, explain how it works and how to use it.

From chapter 6 onwards, instead of saying "now add this code to file x," the code being examined will be reproduced in the text and its location within the sample code indicated.

Key commands to perform particular operations will be reproduced but not at the level of:

```
cd xxxxx
mv yyy zzzzz
etc etc
```

It is therefore important to read this book with the code accessible. While each chapter can be read and will make complete sense without directly referring to the full sample code, it's important to spend time exploring and understanding where each file fits in overall.

At a minimum I'd recommend reading each chapter, then opening the files referenced from the sample repository and making sure their contents and structure make sense before moving onto the next chapter.

It is not necessarily envisaged that you'll continue to use the sample code for your own servers (although please feel free to use it as a starting point). Rather that by the end of the book, you'll be comfortable creating a template of your own, customised to your needs.

Structure of Part 1 (Chef)

Chapters 2 and 3 deal with the type of stack covered in this book and language of chef, servers and provisioning.

Chapter 4 is for any 'read the manual afterwards' types and provides the minimal command set needed to get a fully functional server up and running using the example repository.

Chapter 5 follows a traditional step by step approach to get to grips with the structure and workflow of chef. It then covers the basics of how chef projects are structured using as an example a simple Redis configuration, all the way through to applying this Redis configuration to a VPS.

Chapters 6 - 12 move onto using the rails server template configuration included with this book and available at https://github.com/TalkingQuickly/rails-server-template which can serve as a basis for your production configuration.

Using this template configuration we'll cover:

- How best to install Ruby
- Using Monit to ensure everything runs smoothly without our intervention
- Basic security precautions
- Firewall Management with UFW
- Managing users and public keys
- Creating and populating databases (with your provider of choice)
- Setting up Redis and common gotchas
- Setting up Memcached

Chapter 13 covers how to use Vagrant to test your Chef recipes locally.

Structure of Part 2 (Capistrano)

In Part 2 we'll cover deployment (rather than provisioning), in particular:

- What should be managed by Capistrano v Chef
- Creating simple, modular Capistrano Recipes
- Avoiding falling foul of $PATH
- Dealing with Virtualhost files
- Managing SSL Certificates
- Configuring Unicorn
- Zero Downtime Deployment & Gotchas
- Log Rotation
- Copying databases between environments
- Managing Cron jobs with Whenever
- Managing background jobs with Sidekiq

By the end of the book you will have a solid blueprint you can use whenever you need to provision a new server or deploy a new Rails application.