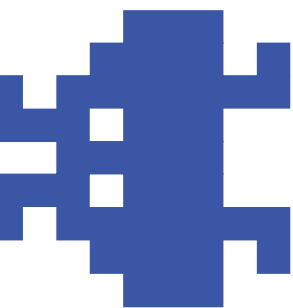
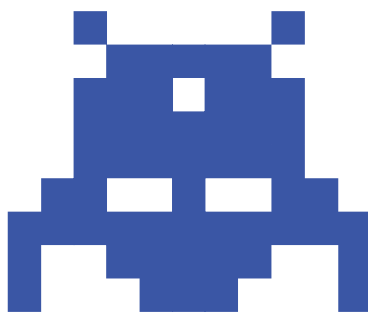
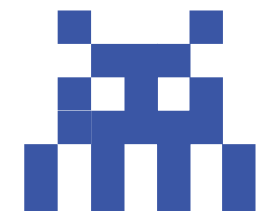


Carlos D. Camacho

# CHEF BY EXAMPLE

OHA!  
CHEFS!



Practical Exercises in a Successful Chef Deployment



# Chef by example

Practical Exercises in a Successful Chef Deployment.

Carlos Camacho



# Contents

<b>Preface</b>	<b>v</b>
0.1 What is DevOps? . . . . .	v
0.2 What this book covers . . . . .	vi
0.3 What you need for this book . . . . .	vi
0.4 Who this book is for . . . . .	vi
0.5 Conventions . . . . .	vi
 <b>1 Getting Started</b>	 <b>1</b>
1.1 Defining the scenario . . . . .	1
1.2 Development process . . . . .	1
1.3 Infrastructure automation . . . . .	1
1.4 Continuous deployments . . . . .	1
1.5 Continuous delivery . . . . .	1
1.6 Maintenance cycle . . . . .	1
 <b>2 Chef</b>	 <b>3</b>
2.1 Chef components . . . . .	4
2.2 Knife . . . . .	4
2.3 OHAI . . . . .	4
2.4 Attributes . . . . .	4
2.5 Databags . . . . .	4
2.6 Environments . . . . .	4
2.7 Lightweight resources and providers . . . . .	4
2.8 Cookbooks . . . . .	4

2.9	Recipes . . . . .	4
2.10	Install CHEF . . . . .	4
2.11	Install the Workstation . . . . .	8
2.12	Installing the Open Source Web Interface . . . . .	8
2.12.1	Requirements installation . . . . .	8
2.12.2	COnfigure the default paramenters . . . . .	9
2.12.3	Use the interface on demand or install it as a service . . . . .	9
<b>3</b>	<b>Development environment</b>	<b>11</b>
3.1	Source repository . . . . .	11
3.2	Package repository . . . . .	11
3.3	CI Server . . . . .	11
<b>4</b>	<b>Deployment environments</b>	<b>13</b>
4.1	Physical or CCloud environments . . . . .	13
4.2	Continous Integration . . . . .	13
4.3	Integration . . . . .	13
4.4	Test . . . . .	13
4.5	Go-Live! . . . . .	13
<b>5</b>	<b>Automating delivery pipelines</b>	<b>15</b>
<b>6</b>	<b>Maintenance cycle</b>	<b>17</b>

# Preface

A Chef Style DevOps Kung-fu implementation, inspired in the Adam Jacob keynote from Chefconf 2015. A practical to work with infrastructure automaion, continuous delivery, continuous deployments and the platform maintenance life cycle.

The source code of this book is hosted in [Github](#), everyone can fork and make pull requests; you are all invited.

## 0.1 What is DevOps?

There isnt an agreed definition for DevOps yet. DevOps is a cultural and professional movement, focused on how we build and operate high velocity organizations, born from the experience of its practitioners. Is a unique approach, based in previous experience and focused on customers.

- Principles (Universal)
  - Based on prioritize people over products over companies.
  - Design for the safety, contentment, knowledge and freedom of both your peers and your customers.
  - Eliminate non-value-added actions and processes.
  - Continuously improve your processes.
  - Adapt to needs.
  - Small improvements over the time.

- Fail faster to learn faster. Be calm, fix things and improve your processes.
  - Workflows automation.
  - Diversity, gets feedback, have different opinions, argue, make demonstrations on your points of view.
  - Demo all the things you have or you are working in.
  - Be the owner of your work, love your work, and find reasons to do your work.
  - Improve and do things right even if is hard (At first).
  - Make decisions based on your experience and proof your point of view.
- Forms (Shared)
  - Applications (Unique)

## **0.2 What this book covers**

## **0.3 What you need for this book**

## **0.4 Who this book is for**

## **0.5 Conventions**



# **Chapter 1**

## **Getting Started**

**1.1 Defining the scenario**

**1.2 Development process**

**1.3 Infrastructure automation**

**1.4 Continuous deployments**

**1.5 Continuous delivery**

**1.6 Maintenance cycle**





## Chapter 2

# Chef

### 2.1 Chef components

### 2.2 Knife

### 2.3 OHAI

### 2.4 Attributes

### 2.5 Databags

### 2.6 Environments

### 2.7 Lightweight resources and providers

### 2.8 Cookbooks

### 2.9 Recipes

### 2.10 Install CHEF

The first step is to check the server host name.

**Listing 2.1**

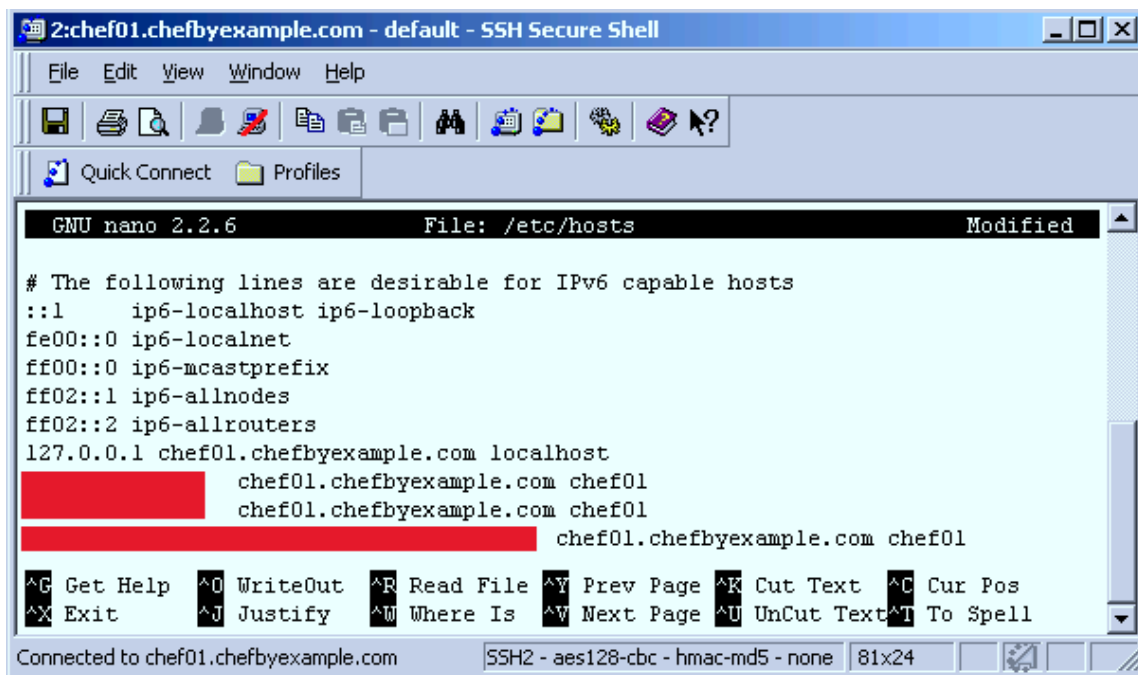
```
root@chef01:~# hostname -f
```

Now we need to edit the host name configuration file in order to add the FQDN of the server.

**Listing 2.2**

```
root@chef01:~# sudo nano /etc/hosts
```

Your configuration file should be something similar to this one.

A screenshot of a terminal window titled "2:chef01.chefbyexample.com - default - SSH Secure Shell". The window shows the GNU nano 2.2.6 text editor editing the file /etc/hosts. The file content includes IPv6 addresses, a mapping for 127.0.0.1 to chef01.chefbyexample.com, and three redacted entries for the FQDN chef01.chefbyexample.com. The terminal window has a menu bar (File, Edit, View, Window, Help) and a toolbar with icons for file operations. At the bottom, it shows the connection status: "Connected to chef01.chefbyexample.com" and "SSH2 - aes128-cbc - hmac-md5 - none" with a window size of 81x24.

```
GNU nano 2.2.6 File: /etc/hosts Modified

# The following lines are desirable for IPv6 capable hosts
::1    ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
127.0.0.1 chef01.chefbyexample.com localhost
[redacted] chef01.chefbyexample.com chef01
[redacted] chef01.chefbyexample.com chef01
[redacted] chef01.chefbyexample.com chef01

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

Connected to chef01.chefbyexample.com SSH2 - aes128-cbc - hmac-md5 - none 81x24
```

Update the packages list and update the server.

**Listing 2.3**

```
root@chef01:~# sudo aptitude update
root@chef01:~# sudo aptitude upgrade
```

Download the latest version of the chef server to the root folder of the current user.

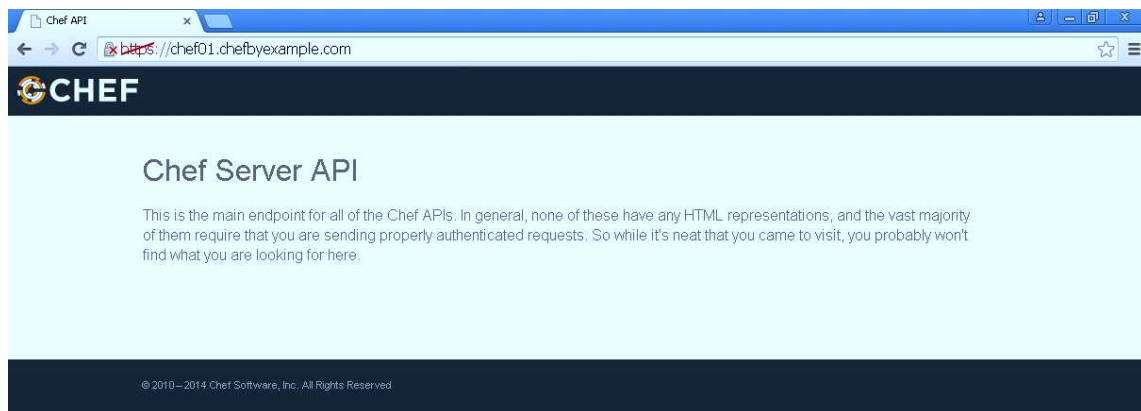
### Listing 2.4

```
root@chef01:~# wget https://web-dl.packagecloud.io/chef/stable/packages/ubuntu/trusty/chef-server
```

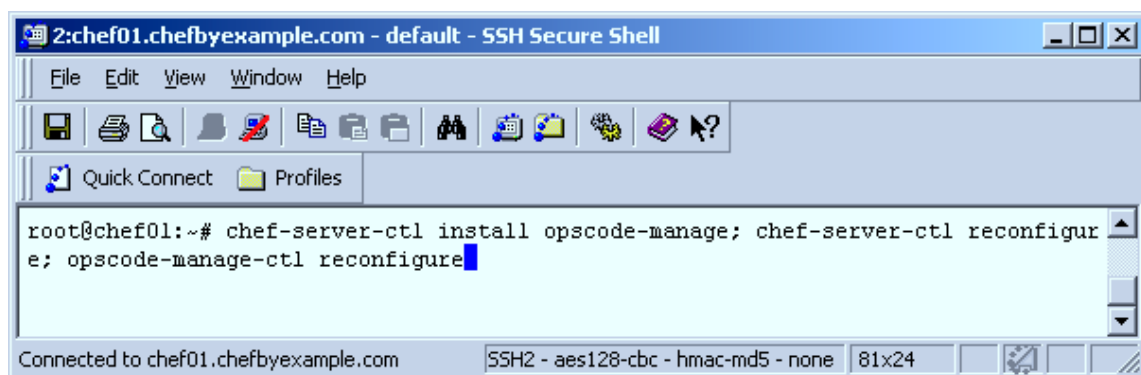
Install the Chef Server.

### Listing 2.5

```
root@chef01:~# sudo dpkg -i chef-server-core_*.deb
root@chef01:~# sudo chef-server-ctl reconfigure
```



This should be the result. . . .  
Install additional modules.



(Premium freature up to 25 nodes..)

### Listing 2.6

```
root@chef01:~# chef-server-ctl install opscore-manage;  
root@chef01:~# chef-server-ctl install opscore-push-jobs-server;  
root@chef01:~# chef-server-ctl install opscore-reporting;  
root@chef01:~# opscore-manage-ctl reconfigure  
root@chef01:~# opscore-push-jobs-server-ctl reconfigure;  
root@chef01:~# opscore-reporting-ctl reconfigure;  
root@chef01:~# chef-server-ctl reconfigure;
```

We need to create the first user (Admin user)

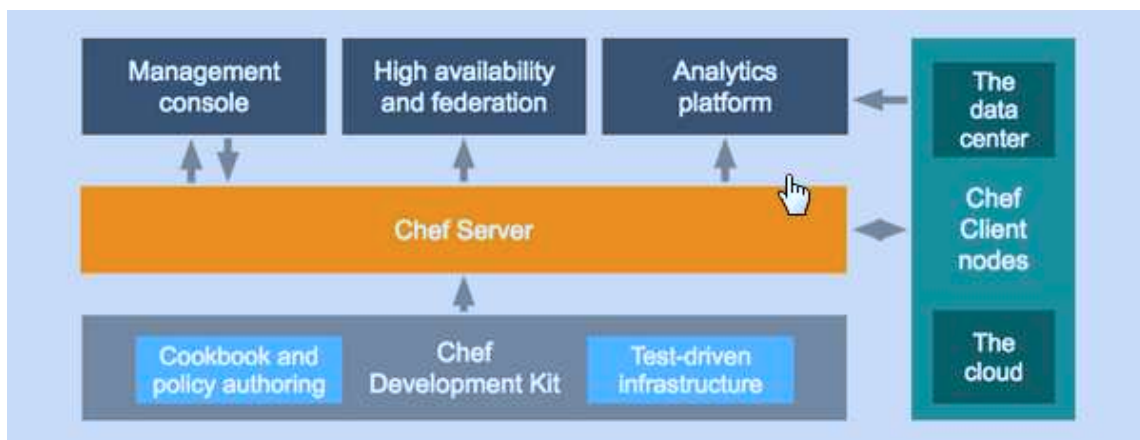
### Listing 2.7

```
root@chef01:~# chef-server-ctl user-create admin the administrator the_good@chefbyexample.com 4
```

WE need to create the first organization and add the first created user to it.

### Listing 2.8

```
root@chef01:~# sudo chef-server-ctl org-create chefbyexample "ChefByExample.com" --association_
```



Now the Chef Server is fully operative, now we need to add the Workstation.

## 2.11 Install the Workstation

### Listing 2.9

```
sudo wget https://opscode-omnibus-packages.s3.amazonaws.com/ubuntu/12.04/x86_64/chefdk_0.7.0-1_
sudo dpkg -i chefdk_*.deb

sudo chef generate repo chef-repo

mkdir ~/chef-repo/.chef
scp root@chef01.chefbyexample.com:/root/admin.pem ~/chef-repo/.chef
scp root@chef01.chefbyexample.com:/root/chefbyexample-validator.pem ~/chef-repo/.chef

nano ~/chef-repo/.chef/knife.rb

current_dir = File.dirname(__FILE__)
log_level      :info
log_location   STDOUT
node_name      "admin"
client_key     "#{current_dir}/admin.pem"
validation_client_name "chefbyexample-validator"
validation_key  "#{current_dir}/chefbyexample-validator.pem"
chef_server_url "https://chef01.chefbyexample.com/organizations/chefbyexample"
syntax_check_cache_path "#{ENV['HOME']}/.chef/syntaxcache"
cookbook_path   [ "#{current_dir}/../cookbooks" ]

knife ssl fetch

Bootstrapping the first node

knife bootstrap node01.chefbyexample.com -N node01
```

## 2.12 Installing the Open Source Web Interface

### 2.12.1 Requirements installation

Installation steps, just run:

### Listing 2.10

```
aptitude install git rubygems1.9.1 ruby1.9.1-dev build-essential
mkdir -p /var/www/
```



```
cd /var/www/  
git clone https://github.com/carlosdcg/chef-server-webui  
cd chef-server-webui  
gem install bundler  
bundle install
```

### 2.12.2 COnfigure the default paramenters

Configure the web app in `/var/www/chef-server-webui/config/application.rb`

#### Listing 2.11

```
config.chef_server_url = "http://127.0.0.1"  
config.rest_client_name = "pivotal"  
config.rest_client_key = "/etc/opscode/pivotal.pem"  
config.admin_user_name = "admin"  
config.admin_default_password = "4dmln1str4t0r"  
config.rest_client_custom_http_headers = {}  
#This app only supports one organization, like the Open Source Chef Server 11  
config.default_organization = "organizations/chefbyexample/"
```

### 2.12.3 Use the interface on demand or install it as a service

Once the Web UI is installed, from `/var/www/chef-server-webui` run:

To test in the default port 9292:

#### Listing 2.12

```
rackup config.ru
```

To run as a daemon in another port to test:

#### Listing 2.13

```
rackup config.ru -D -p 1234
```

Once you have tested it, to create the init scripts and install the run levels—

**Listing 2.14**

```
#TO remove the script from the default run-levels
#sudo update-rc.d -f chef-server-webui remove
sudo chmod 755 /var/www/chef-server-webui/init/chef-server-webui.sh
ln -s /var/www/chef-server-webui/init/chef-server-webui.sh /etc/init.d/chef-server-webui
sudo chmod 755 /etc/init.d/chef-server-webui
sudo chown root:root /etc/init.d/chef-server-webui
sudo update-rc.d chef-server-webui defaults
```

## **Chapter 3**

# **Development environment**

**3.1 Source repository**

**3.2 Package repository**

**3.3 CI Server**



## **Chapter 4**

# **Deployment environments**

**4.1 Physical or CLOUD environments**

**4.2 Continuous Integration**

**4.3 Integration**

**4.4 Test**

**4.5 Go-Live!**



## **Chapter 5**

# **Automating delivery pipelines**





# Chapter 6

## Maintenance cycle

- General configuration structure for the chef-repo:
  - chef-repo/environments/banana.rb
  - chef-repo/environments/potato.rb
  - chef-repo/environments/kiwi.rb
  - chef-repo/data\_bags/banana.rb
  - chef-repo/data\_bags/potato.rb
  - chef-repo/data\_bags/kiwi.rb:
  - chef-repo/roles/base.rb
  - chef-repo/roles/web.rb
  - chef-repo/roles/db.rb
- Banana cookbook structure:
  - chef-repo/cookbooks/banana/templates/default/\*.erb
  - chef-repo/cookbooks/banana/attributes/default.rb
- Potato cookbook structure:
  - chef-repo/cookbooks/potato/templates/default/\*.erb

- chef-repo/cookbooks/potato/attributes/default.rb

- Kiwi cookbook structure: \* chef-repo/cookbooks/kiwi/templates/default/\*.erb
  - \* chef-repo/cookbooks/kiwi/attributes/default.rb

- A node belongs to an environment in which case, will override the default configuration per the corresponding one.

- Override app attributes for kiwi (Non sensitive info) Override app attributes for kiwi (Sensitive info) Define the recipes for the xxx role

- Default configuration templates for kiwi Default configuration values according the templates for kiwi