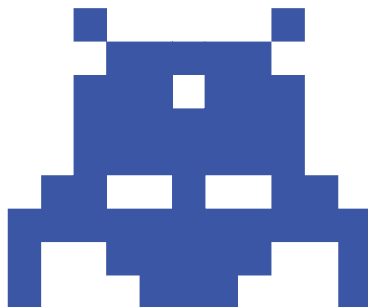
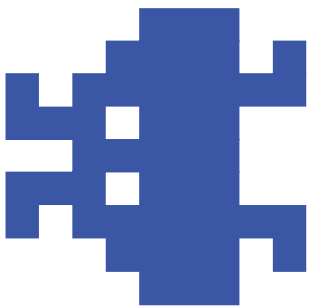
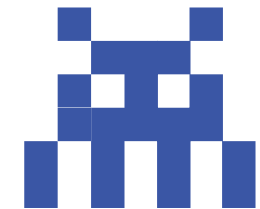


Carlos D. Camacho

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

Preface	v
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
 1 Getting Started	 1
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
 2 Chef	 3
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 one Workstation	9
2.12	Install	10
2.13	Use	11
3	Development environment	13
3.1	Source repository	13
3.2	Package repository	13
3.3	CI Server	13
4	Deployment environments	15
4.1	Physical or CCloud environments	15
4.2	Continuous Integration	15
4.3	Integration	15
4.4	Test	15
4.5	Go-Live!	15
5	Automating delivery pipelines	17
6	Maintenance cycle	19

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

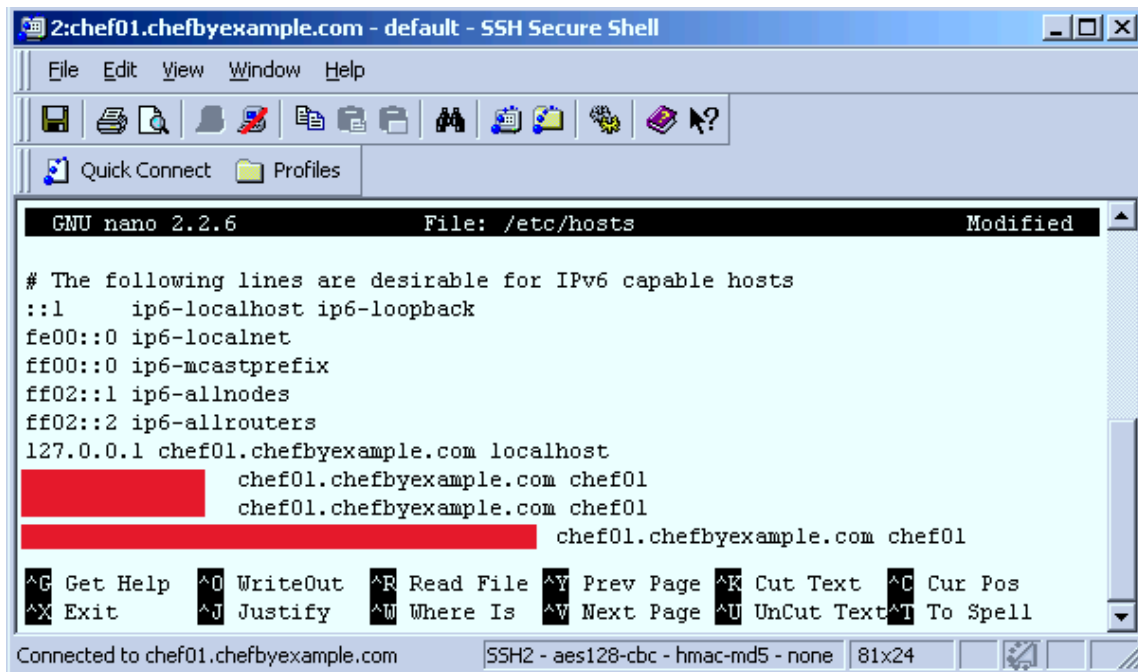
2.9 Recipes

2.10 Install CHEF



Listing 2.1

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



The screenshot shows a terminal window titled "2:chef01.chefbyexample.com - default - SSH Secure Shell". The window contains the GNU nano 2.2.6 editor editing the file /etc/hosts. The file content is as follows:

```
# 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
        chef01.chefbyexample.com chef01
        chef01.chefbyexample.com chef01
        chef01.chefbyexample.com chef01
```

The bottom status bar of the terminal window indicates: "Connected to chef01.chefbyexample.com", "SSH2 - aes128-cbc - hmac-md5 - none", and "81x24".

Listing 2.2

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



The screenshot shows a terminal window titled "2:chef01.chefbyexample.com - default - SSH Secure Shell". The terminal displays the command:

```
root@chef01:~# wget https://web-dl.packagecloud.io/chef/stable/packages/ubuntu/trusty/chef-server-core_12.2.0-1_amd64.deb
```

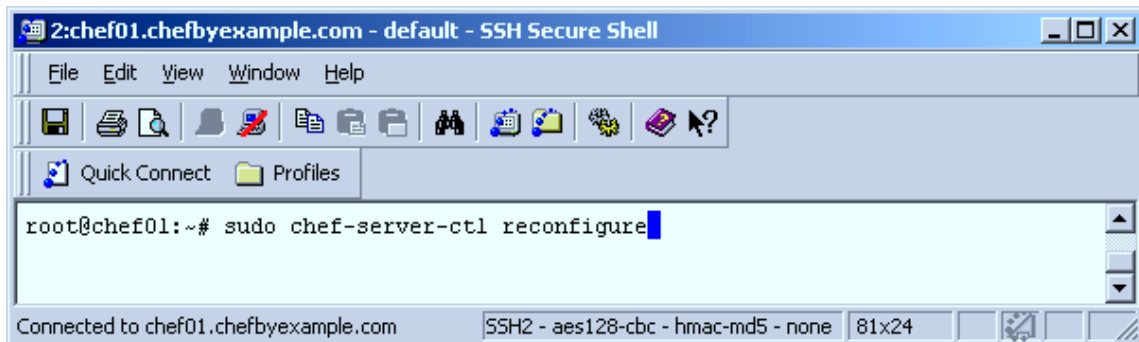
The bottom status bar of the terminal window indicates: "Connected to chef01.chefbyexample.com", "SSH2 - aes128-cbc - hmac-md5 - none", and "81x24".

Listing 2.3

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

**Listing 2.4**

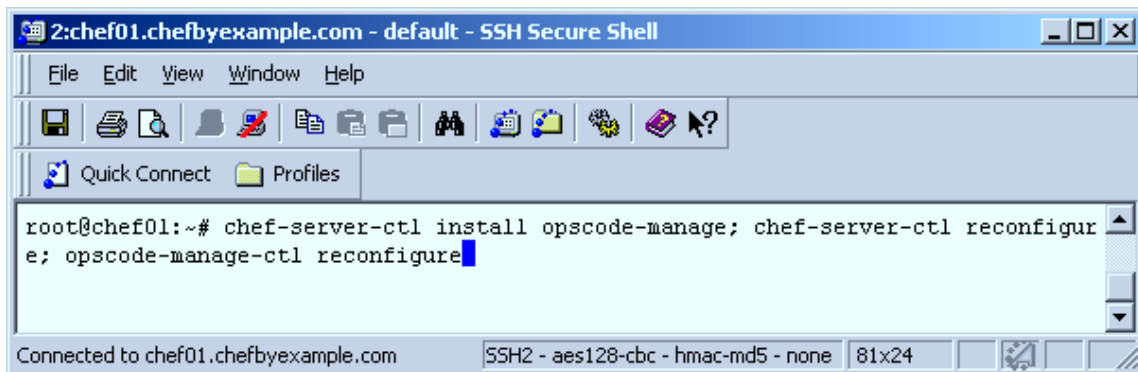
```
root@chef01:~# sudo dpkg -i chef-server-core_*.deb
```

**Listing 2.5**

```
root@chef01:~# sudo chef-server-ctl reconfigure
```



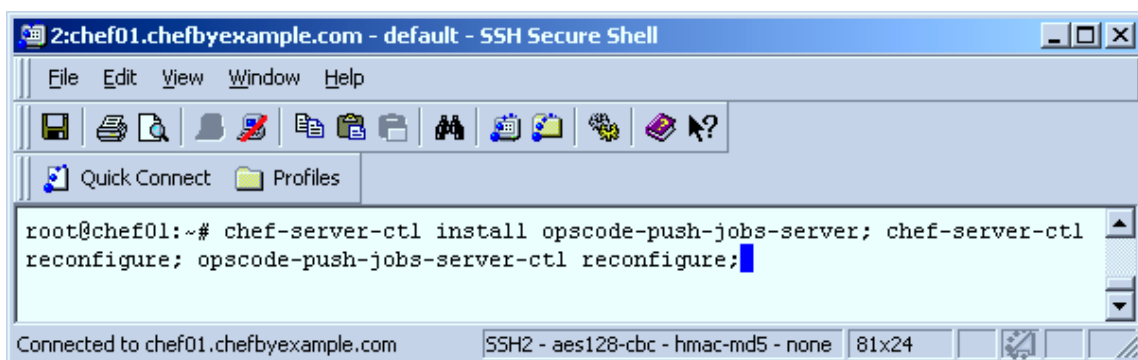
Web interface



(Premium freature up to 25 nodes..)

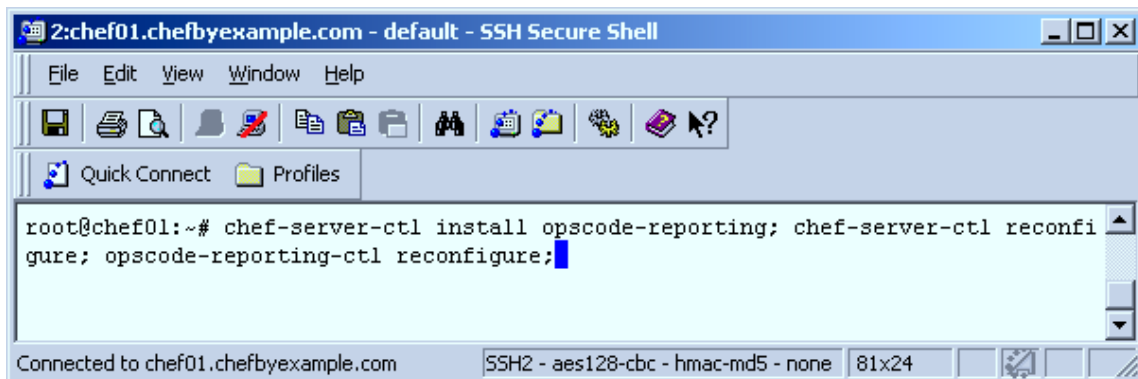
Listing 2.6

```
root@chef01:~# chef-server-ctl install opscode-manage; chef-server-ctl reconfigure; opscode-man
```



Listing 2.7

```
root@chef01:~# chef-server-ctl install opscode-push-jobs-server; chef-server-ctl reconfigure; o
```

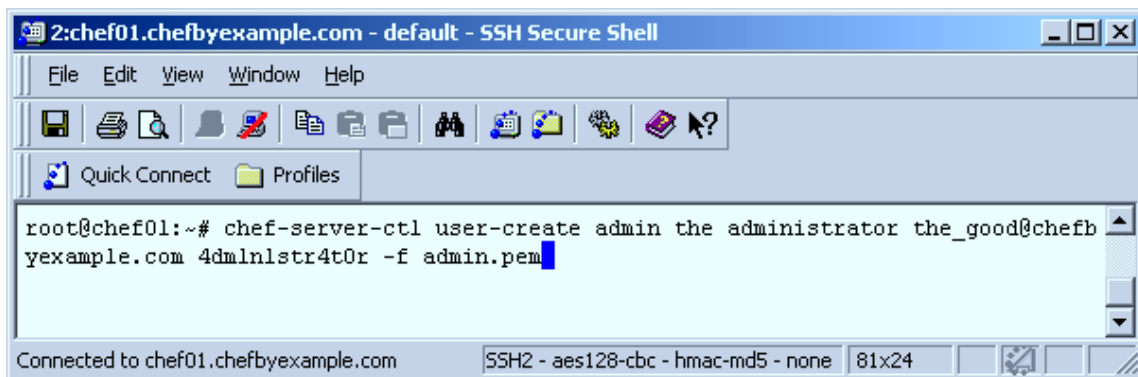


The screenshot shows an SSH terminal window titled "2:chef01.chefbyexample.com - default - SSH Secure Shell". The terminal displays the command `chef-server-ctl install opscode-reporting; chef-server-ctl reconfigure; opscode-reporting-ctl reconfigure;` being entered. The window includes a menu bar (File, Edit, View, Window, Help), a toolbar with various icons, and a status bar at the bottom indicating the connection details: "Connected to chef01.chefbyexample.com", "SSH2 - aes128-cbc - hmac-md5 - none", and "81x24".

(Premium freature up to 25 nodes..)

Listing 2.8

```
root@chef01:~# chef-server-ctl install opscode-reporting; chef-server-ctl reconfigure; opscode-
```



The screenshot shows an SSH terminal window titled "2:chef01.chefbyexample.com - default - SSH Secure Shell". The terminal displays the command `chef-server-ctl user-create admin the administrator the_good@chefbyexample.com 4dmlnlstr4t0r -f admin.pem` being entered. The window includes a menu bar (File, Edit, View, Window, Help), a toolbar with various icons, and a status bar at the bottom indicating the connection details: "Connected to chef01.chefbyexample.com", "SSH2 - aes128-cbc - hmac-md5 - none", and "81x24".

Listing 2.9

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



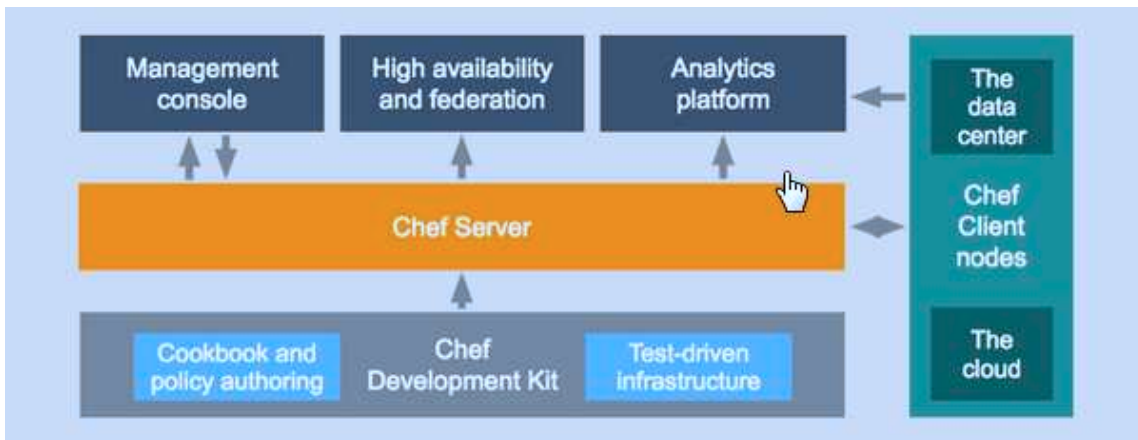
```

root@chef01:~# sudo chef-server-ctl org-create chefbyexample "ChefByExample.com"
--association user admin -f chefbyexample-validator.pem

```

Listing 2.10

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



Chef Server installed..

2.11 Install one Workstation

Listing 2.11

```

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 Install

Installation steps, just run:

Listing 2.12

```

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

```

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

Listing 2.13

```
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 = "4dm1n1str4t0r"
config.rest_client_custom_http_headers = {}
#This app only supports one organization, like the Open Source Chef Server 11
config.default_organization = "organizations/chefbyexample/"
```

2.13 Use

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

To test in the default port 9292:

Listing 2.14

```
rackup config.ru
```

To run as a daemon in another port:

Listing 2.15

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


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