Chef

Saving Time (and Money) With Automated Provisioning

Trotter Cashion

Hoopla Software

About Me

@cashion

http://github.com/trotter

Where I've Been

Eastmedia (2006)

Motionbox (2006-2009)

Algorithmics (2009-2010)

Hoopla (2010-?)

Our Agenda

American car companies

The problem with provisioning

Using automation to solve the problem

Chef as the automation tool

A Brief History

oldcar

Early Cars

Extremely Expensive ($2-3k)

Hand Crafted

12.5 hours

Ford

Model T - $850

Heavily Automated

1.5 hours

Scary Statistic

250 early car companies failed by 1930

(This stat came from wikipedia, ymmv)

"Most tech companies are essentially American auto companies pre-Ford."

- Me

The Problem Today

Manual!

The Artisan Sysadmin

mysql, postgres, ...

apache, nginx, ...

memcached, redis, mongodb, riak ...

iptables, sshd, ...

What You Get

Snowflakes

Heisenbugs

No More Cash

Automate Or Die

Tenderizer

First box took 1 week

Second box took 2 days

Third box took 1/2 a day

After that it was gravy

Architecture

bash

static resource files

totally f#ck|ng awesome

But It Sucks

Difficult to maintain

Adding new machines was time consuming

Difficult to reprovision old machines

Slow

Back to the Drawing Board

Chef!

Attributes

Written by Opscode

Written in Ruby

Ruby DSL

What Makes It Awesome

Ruby DSL

Cookbooks

Mostly Idempotent

Keeps machine and code in lock-step

Two Flavors

Chef-Solo

Chef Server / Chef Client

Today...

We talk Chef Solo

Vs. Tenderizer

First box took me 2 days

Next box took 1/2 hour

Initial provisioning ~15-30 minutes

Typical runtime is < 2 minutes

That's like Unicorns and shit.

Getting Started

On your local machine

```
$ git clone http://github.com/opscode/chef-repo
$ gem install spatula
$ spatula prepare db-server.yourcompany.com
$ spatula install mysql
```

spatula is in flux, so don't be surprised if commands have changed.

Make a node config

```
// config/db_server.json

{
   "mysql": {
      "server_root_password": "abcdefg
   },
   "recipes": ["mysql"]
}
```

Again Locally

```
$ cp config/solo.rb.example config/solo.rb
$ spatula cook db-server.yourcompany.com db_server
```

spatula is in flux, so don't be surprised if commands have changed.

ALL DONE!

Under the hood

Simple MySQL recipe

```
# cookbooks/mysql/recipes/default.rb
install_package "mysql-devel"
install_package "mysql-client"
install_package "mysql-server"
```

Simple Apache Virtual Host

```
<VirtualHost *:80>
  ServerName <%= @params[:server_nam
  DocumentRoot <%= @params[:docroot]
  RewriteEngine On

  <Directory <%= @params[:docroot] %
    Options FollowSymLinks
    AllowOverride None
    Order allow,deny
    Allow from all
```

Architecture

Key Components

Nodes

Roles

Cookbooks

The Chef Directory

config/

cookbooks/

roles/

site-cookbooks/ (sometimes)

A Sample Role

```
# roles/webserver.rb

name            "webserver"
description "The base role for syste
            " that serve HTTP traffi
run_list    "recipe[apache2]",
            "recipe[apache2::mod_ssl
            "role[monitor]"
default_attributes "apache2" =>
  { "listen_ports" => [ "80", "443"
```

The Cookbook Directory

recipes/

files/

templates/

attributes/

A Cookbook

metadata.rb

definitions/

libraries/

Top 4 Resources

Template

Directory

RemoteFile

Deploy

Learn More

Other Provisioning Frameworks

Cfengine - http://www.cfengine.org/

Puppet - http://www.puppetlabs.com/

More on Chef

http://wiki.opscode.com/display/chef/Home

http://cookbooks.opscode.com/

http://wiki.opscode.com/display/chef/Resources

Remember

Car Companies

Automate or Die!

Use Chef!

Contact Me

twitter - @cashion

github - http://github.com/trotter

email - cashion on the gmailz

Thank You