

Presenting Chef Solo

Tell me again: Why didn't you automate that?

@kyleburton
kyle.burton@gmail.com

Prepared Especially for: CPLUG
September 14th 2010

whoami

- Head of Technology @ Relay Network
- Consumer \Leftrightarrow Billers
- I Love Open Source

(and don't begrudge commercial software)



What's my Goal?

Should I...

- Teach you how to write a recipe?
- Teach you how to run chef?
- Contrast with CFEngine, Puppet, etc.?

For the love of Pete:
Automate!

Why: Repeatability

- AKA: Configuration Management
- Not just for ‘Software’
- “But I’m only going to do this once”
(what a crok...)

Why: A Record

- Forces you to plan, to write down what you expect to happen.
- Can now start to measure and document the unexpected
- Lets you see where you can introduce checkpoints and recovery plans
- You can see exactly what you did..months from now

Why: Good Habit

- You brush your teeth every day,
it's a good habit.
- All of this gets easier if you just do it.
- Practice is worth more than I can tell
you.

Why: Scale Yourself

- Improve your ROI
- Smaller Teams (companies usually like this)
- Free up your time spent on repetitive tasks
- *Increase the size of your...silicon empire*

How did I Get Here?

- ‘Traditional’ Software Developer
- Joined Highly Skilled Team: had to set up environment w/no docs
- Demo Day + Hosting Provider Issues
- Joined Relay This Year: 1, 2, 1, 2

Enough!

What is Chef?

- A tool for Devops
- A Framework

WTF is Devops?

- Blurring of Development and Operations
- Operations == Traditional IT
- Great Sys Admins already knew this
- Simply the idea that the dichotomy wasted resources, time, lead to inefficiencies
- Brought on by the move to virtualization, to the cloud.

Tool: Chef Solo

- Cookbooks, Recipes, Resources, Nodes, Roles, Providers, Templates
- Confession: I am partway through my journey with Chef.

What For?

- Developer Workstations
- Staging and Testing Environments
- I now get a scary feeling when I work on systems that I haven't used these kinds of tools to set up - I know I can't reproduce their state (crap!)
- Production Systems

Getting Started

- You won't do it 'right' the first time.
- Who the F*&%!K cares?
- I'm not doing it 'right' (things have changed since I started, plus I'm still learning)
- Do it anyway, you'll learn, use what you learn to improve as you go.
- Practice, above all, Practice

How I Got Started (mostly)

- `gem install chef`
- `git clone git://github.com/opscode/chef-repo.git`
- `cd chef-repo`
- `rm -rf cookbooks`
- `git clone git://github.com/opscode/cookbooks.git`

How I Got Started (mostly)

- `mkdir site-cookbooks`
- `cp -r cookbooks/postgresql/ \`
`site-cookbooks/my-recipe`
- modify the recipe

How You Should Get Started

- Listen to this, but keep some perspective...
- Read The Current Introduction
- Learn Knife
- Don't follow my example too closely

Running Chef Solo

- Create a node config: config/app-server.json
- rsync or scp -r the whole chef-repo to a node
- ssh to the node and run:
 - `chef-solo -c config/solo.rb \`
`-j config/app-server.json`

Node Configuration

```
{ "snapclean": {  
  "users": [ { "name": "kyle" },  
             { "name": "snapclean" },  
             ... },  
  "recipes": [  
    "snapclean-base",  
    "snapclean-nginx",  
    "snapclean-unicorn"  
  ]  
}
```

Recipes

- Gotcha!
 - Names: not just a string
 - Declarative: what runs and when
- Using Standard Recipes
 - Overriding Defaults
- Creating Your Own

Resources

- Files
- Templates (ERB)
- Directories
- Scripts (bash, ruby)
- Packages (apt-get)

recipes/default.rb

- show some examples and best practices here
- Talk about ohai(?)
- How configuration presents it self (node global)
- Running Scripts
- Installing Packages
- Directories, Templates

templates/default/foo.config

- Show an example and discuss best practices
- Passing Parameters
- Accessing Node Configuration

What I do...

- `vi /etc/ssh/sshd_config`
- ...did it work?
- `scp root@host:/etc/ssh/sshd_config \`
`site-cookbooks/prod-sshd/templates/default/`
- `vi site-cookbooks/prod-sshd/recipes/default.rb`
(add file/template declaration)
- Incremental enhancements as I need more configuration points

Managing

- Your Cookbooks and Recipes
- Develop Shared Libraries
- Keep an eye on when to make something configurable vs making things too flexible to maintain.
- Use the guards: `:if` and `:not_if`

Automate Please

- Get Started
- Share your Experiences
- Continually Improve

Good Luck!

Luck Favors the Paranoid, I mean Prepared
(practice).

Thank You!

References

- <http://wiki.opscode.com/display/chef/Home>