

Chef



What is chef?

Chef is a Ruby DSL (domain specific language) for configuring GNU/Linux (or BSD) machines as servers, eg: Ruby on Rails Server.

There are 2 flavours "Chef Server" and "Chef Solo".

I choose Chef Solo as I'm just setting up one server at a time and it's easier to get started.

Whet is cheff?

Cheff is a Rooby DSL (dumeeen speceeffic lungooege-a) fur cunffeegooring GNU/Leenoox (oor BSD) mecheenes es serfers, ig: Rooby oon Reeels Serfer.

Zeere-a ere-a 2 flefuors "Cheff Serfer" und "Cheff Sulu".
I chuuse-a Cheff Sulu es I'm joost setteeng up oone-a serfer et a teeme-a und it's ieseeer tu get sterted.
Bork Bork Bork!

Translated using debian package 'filters' and the 'chef' binary.

Why use chef?

Easy...

once setup

Quick

Headless, unattended, optimised

Freedom

Change providers, distros, versions

Helpful tools

VirtualBox

Short script to restore & restart VirtualBox

[apt_cacher/squid](#)

Test server setup

Based on a standard VPS offering.

1. Install Ubuntu in VirtualBox

Download from Ubuntu Server 10.04 LTS

Install in Virtualbox

Select OpenSSH whilst installing

2. Setup requirements

Passwordless sudo

/etc/sudoers file:

```
<your user> ALL=(ALL) NOPASSWD: ALL
```

3. VirtualBox bridged network

<http://www.virtualbox.org/manual/ch06.html>

Within VirtualBox

- Open Virtualbox
- Click on my server
- Click settings
- Click Network
- Select 'Bridged Adapter' from the 'Attached to' select box
- In the 'Name' select box select your hosts active network adapter
- Click OK

4. Cache (squid)

Squid.conf should have these lines:

```
cache_replacement_policy heap LFUDA
```

```
maximum_object_size 100 MB
```

```
refresh_pattern Packages\.bz2$ 0 20% 4320 refresh-ims
refresh_pattern Sources\.bz2$ 0 20% 4320 refresh-ims
refresh_pattern Release\.gpg$ 0 20% 4320 refresh-ims
refresh_pattern Release$ 0 20% 4320 refresh-ims
refresh_pattern . 0 20% 4320 refresh-ims
```

4. Cache (apt-cacher)

Install it via apt-get

```
$ sudo apt-get install apt-cacher apache2
```

Make it start at boot time

```
$ sudo vi /etc/default/apt-cacher
```

Change Autostart to 1

Restart services

```
$ sudo service apache2 restart
```

```
$ sudo service apt-cacher start
```

Change sources

from

```
deb http://example.debian.org/debian unstable main contrib non-free
```

to

```
deb http://<hostname>:3142/example.debian.org/debian unstable main contrib non-free
```

5. VirtualBox restart script

Quickly restore & restart your VirtualBox

```
#!/bin/bash
# Reset and restart my server as a headless server.
#
# Control my Virtualbox machine
VBoxManage controlvm Ubuntu poweroff
VBoxManage snapshot Ubuntu restorecurrent
VBoxManage startvm --type headless Ubuntu
```

Development Machine

Setup requirements

Password ssh Make sure you can access your server without a password

```
$ ssh-copy-id <user>@<host>
```

Bootstrapping

Two scripts are used to boot strap the server

deploy.sh - Compress & copy over chef-solo directory

install.sh - Install just enough to get chef running

The fun part!

Most challenging was the installation of Ruby & Rubygems on Ubuntu 10.04.

Why?

- packages are old
- compiling is crazy
- rvm (Ruby Version Manager) as a user has issues with chef
- rvm as system wide install seems wrong but it works

I choose rvm as system wide!

Config files

solo.rb Sets paths and log levels for chef.

```
root = File.absolute_path(File.dirname(__FILE__))
```

```
file_cache_path root
```

```
cookbook_path root + '/cookbooks'
```

```
role_path root + "/roles"
```

```
log_level :info
```

```
log_location STDOUT
```

Config files

solo.json Set recipes & roles to use.

```
{  
  "run_list": [  
    "role[web_server]",  
    "recipe[pais::default]"  
  ]  
}
```

Running chef-solo

```
$ ./deploy.sh <server>
```

Cookbooks

Cookbooks are filled with recipes on how to make software.

Advantages of cookbooks

- DSL (Domain Specific Language) helps minimise pain.
- Coded to share
- Many cookbooks are online
- Copy a cookbook and add it to your config.
- Make your own cookbook

Create your own cookbook

```
$ mkdir -p cookbooks/mick/recipes
```

Add to 'cookbooks/mick/recipes/default.rb'

```
# System tools  
package 'htop'
```

Add to config

```
{  
  "run_list": ["recipe[mick::default]"]  
}
```


3rd party cookbooks

Find a cookbook on github

<https://github.com/opscode/cookbooks>

Uncompress into cookbooks directory of your project

Add to your run_list

```
{  
  "run_list": ["recipe[apache]"]  
}
```

Attributes

Some cookbooks require attributes, these can be set in your config.

solo.json

```
{
"apache": { "listen_ports": ["81", "8080"] },
"postgresql": { "version": ["8.0"] },
"run_list": [
"recipe[apache]",
"recipe[postgresql::server]"
]
}
```

Roles (optional)

Allows you to group a set of recipes together in a role.

EG: a servers role maybe a 'web server'.

You can call multiple roles in your config (solo.json).

Duplicates are removed automatically (eg: two roles have the same package, it's only installed once).

References

My github

<https://github.com/map7/chef-solo>

Cookbooks

<https://github.com/opscode/cookbooks>

Tutorial from OP

<http://opinionated-programmer.com/2011/06/chef-solo-tutorial-managing-a-single-server-with-chef/>