



Installation & Setup

Chef Installation

Install Chef on your server using the following commands:

```
sudo apt-get update
sudo apt-get install curl
```

```
curl -L https://omnitruck.chef.io/install.sh |
sudo bash
```

Verify the installation:

```
chef-solo -v
```

Setting Up a Cookbook

Download and extract a cookbook repository:

```
wget http://github.com/chef-cookbooks/chef-
repo/tarball/master -O - | tar xzf - --strip-
components=1
```

Knife Tool

Download cookbooks from the supermarket:

```
knife supermarket download mysql
```

Basic Usage

Chef Solo Invocation

Run Chef Solo with a configuration file and JSON attributes:

```
chef-solo -c solo.rb -j web.json
```

solo.rb: Chef Solo configuration file.

web.json: JSON file containing node attributes.

Key Chef Concepts

Resources	Represent a desired state for a part of the system (e.g., a package, a file, a service).
Recipes	Collections of resources that define a configuration policy.
Attributes	Define specific settings and data used by resources.
Cookbooks	Packages that contain recipes, attributes, and other related files.

Resource Examples

Compile From Source

```
execute "tar --no-same-owner -zxf hi.tar.gz"
do
  cwd "/usr/local/src"
  creates "/usr/local/src/node-v#{version}"
end
```

```
bash "compile" do
  cwd "/usr/local/src/node-v#{version}"
  code %[PATH=/usr/local/bin:$PATH ./configure
make]
  creates "/usr/local/src/node-v#
{version}/node"
end
```

Remote File Resource

```
remote_file "/usr/local/src/hi.tar.gz" do
  source "http://..."
  checksum "ab83be..."
  mode 0644
  action :create_if_missing
end
```

Ruby Block

```
ruby_block "name" do
  block { File.read ... }
  not_if { File.exists?(...) }
end
```

Execute Resource

```
execute "name" do
  cwd "..."
  environment({ "PATH" => "..." })
  command "make install"
  creates "..."
end
```

Advanced Configuration

Resource Conditions

```
creates "/usr/local/src/node-v#{version}/node"
not_if { File.exists?('...') }
```

File Resource

```
file '/path/to/file' do
  content 'This is the file content'
  owner 'www-data'
  group 'www-data'
  mode '0644'
  action :create
end
```

Package Resource

```
package 'apache2' do
  action :install
end
```

Service Resource

```
service 'apache2' do
  action [ :enable, :start ]
end
```