

chef-client

Applying Recipes from Multiple Cookbooks

chef-client



```
$ sudo chef-client --local-mode RECIPE_FILE
```

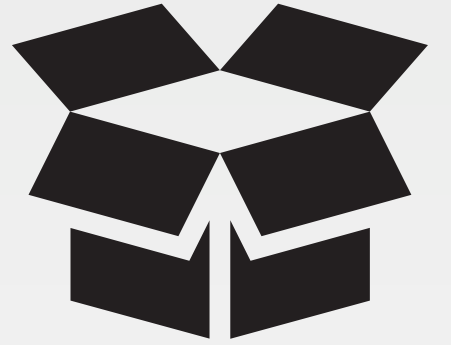
How would we apply both the workstation's setup recipe and apache's server recipe?

Objectives

After completing this module, you should be able to use chef-client to:

- Locally apply multiple cookbooks' recipes with chef-client.
- Include a recipe from within another recipe.

CONCEPT

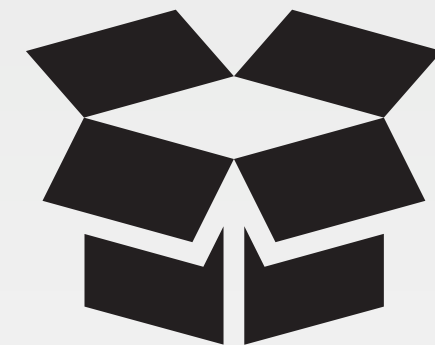


--local-mode

chef-client's default mode attempts to contact a Chef Server and ask it for the recipes to run for the given node.

We are overriding that behavior to have it work in a local mode.

CONCEPT



```
--runlist "recipe[COOKBOOK::RECIPE]"
```

```
-r "recipe[COOKBOOK::RECIPE]"
```

In local mode, we need to provide a list of recipes to apply to the system. This is called a **run list**. A run list is an ordered collection of recipes to execute.

Each recipe in the run list must be addressed with the format **recipe[COOKBOOK::RECIPE]**.

GL: Apply the Cookbook Recipe Locally



```
$ sudo chef-client --local-mode -r "recipe[apache::server]"
```

```
[2016-09-15T14:54:45+00:00] WARN: No config file found or specified on command line, using command line options.
```

```
Starting Chef Client, version 12.13.37
```

```
resolving cookbooks for run list: ["apache::server"]
```

```
Synchronizing Cookbooks:
```

```
- apache
```

```
Compiling Cookbooks...
```

```
Converging 3 resources
```

```
Recipe: apache::server
```

- * yum_package[httpd] action install (up to date)
- * file[/var/www/html/index.html] action create (up to date)
- * service[httpd] action enable (up to date)

GL: Apply the Cookbook Recipe Locally



```
$ sudo chef-client --local-mode -r "recipe[workstation::setup]"
```

```
[2016-09-15T15:15:26+00:00] WARN: No config file found or specified on command line, using command line options.
```

```
Starting Chef Client, version 12.13.37
```

```
resolving cookbooks for run list: ["workstation::setup"]
```

```
Synchronizing Cookbooks:
```

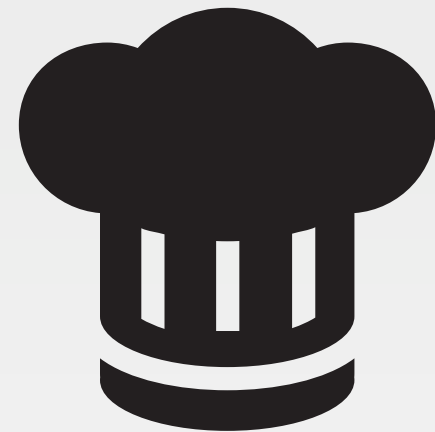
```
- workstation (0.1.0)
```

```
Compiling Cookbooks...
```

```
Converging 2 resources
```

```
Recipe: workstation::setup
```

- * yum_package[tree] action install (up to date)
- * file[/etc/motd] action create (up to date)



Applying a Run List

Using a run list will allow us to specify things more 'logically' instead of with paths.

Objective:

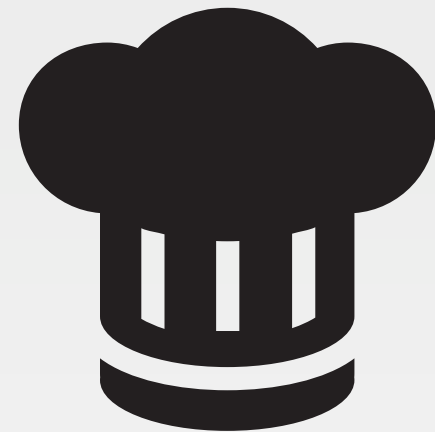
- ✓ Individually apply the apache cookbook's server recipe and workstation cookbook's setup recipe
- ❑ Apply both the apache cookbook's server recipe and workstation cookbook's setup recipe

GL: Apply Both Recipes Locally



```
$ sudo chef-client --local-mode \  
-r "recipe[apache::server],recipe[workstation::setup]"
```

```
[2016-09-15T15:17:27+00:00] WARN: No config file found or specified on  
command line, using command line options.  
Starting Chef Client, version 12.13.37  
resolving cookbooks for run list: ["apache::server","workstation::setup"]  
Synchronizing Cookbooks:  
  - apache  
  - workstation  
Compiling Cookbooks...  
  
Running handlers:  
...
```



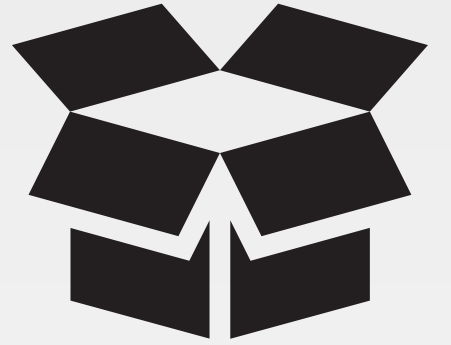
A Succinct Run List

The cookbook only has one recipe that we care about. Could we set that up as the default?

Objective:

- ☐ Load the workstation cookbook's setup recipe in the default recipe
- ☐ Apply the workstation cookbook's default recipe
- ☐ Commit the changes

CONCEPT



-r "recipe[COOKBOOK(::default)]"

When you are referencing the default recipe within a cookbook you may optionally specify only the name of the cookbook.

chef-client understands that you mean to apply the default recipe from within that cookbook.

GL: Apply Recipe using default



```
$ sudo chef-client --local-mode -r "recipe[apache]"
```

```
[2016-09-15T15:17:27+00:00] WARN: No config file found or specified on  
command line, using command line options.
```

```
Starting Chef Client, version 12.13.37
```

```
resolving cookbooks for run list: ["apache::default"]
```

```
Synchronizing Cookbooks:
```

```
- apache
```

```
Compiling Cookbooks...
```

```
Running handlers:
```

```
...
```

DOCS

include_recipe



A recipe can include one (or more) recipes located in cookbooks by using the `include_recipe` method. When a recipe is included, the resources found in that recipe will be inserted (in the same exact order) at the point where the `include_recipe` keyword is located.

<https://docs.chef.io/recipes.html#include-recipes>

GL: The Default Recipe Includes the Setup Recipe

 `~/cookbooks/apache/recipes/default.rb`

```
#  
# Cookbook Name:: apache  
# Recipe:: default  
#  
# Copyright (c) 2016 The Authors, All Rights Reserved.  
  
include_recipe 'apache::server'
```

GL: Apply the Cookbook's Default Recipe



```
$ sudo chef-client --local-mode -r "recipe[apache]"
```

```
WARN: No config file found or specified on command line, using command line options.
```

```
Starting Chef Client, version 12.13.37
```

```
resolving cookbooks for run list: ["workstation"]
```

```
Synchronizing Cookbooks:
```

```
- apache (0.1.0)
```

```
Compiling Cookbooks...
```

```
Converging 2 resources
```

```
Running handlers:
```

```
Running handlers complete
```

```
Chef Client finished, 0/2 resources updated in 12 seconds
```

Lab: 60 minutes



<https://github.com/shekhar2010us/chef-essentials-repo-15/blob/master/labs/chapter%204.md>

Discussion



Why would you want to apply more than one recipe at a time?

What are the benefits and drawbacks of using "include_recipe" within a recipe?

Do default values make it easier or harder to learn?

Q&A



What questions can we help you answer?

- chef-client
- local mode
- run list
- include_recipe



CHEF™