Cookbook Attributes, Attribute Files and Dependencies

Setting Attributes within a Cookbook and Applying the company_web Cookbook to Your Nodes

CHEF

Objectives

After completing this module, you should be able to:

- Explain where cookbook attributes reside
- Create a wrapper cookbook
- Configure dependencies between cookbooks
- Generate a new policyfile and policyfile.lock.json
- Upload the new policyfiles to Chef Infra server and converge the nodes

Attribute Files

The Node object contains many automatic attributes generated by OHAI.

You can also maintain attributes within a cookbook.

These are like variables or parameters for your cookbook and allow recipes to be data driven.

https://docs.chef.io/attributes.html

Best Practices

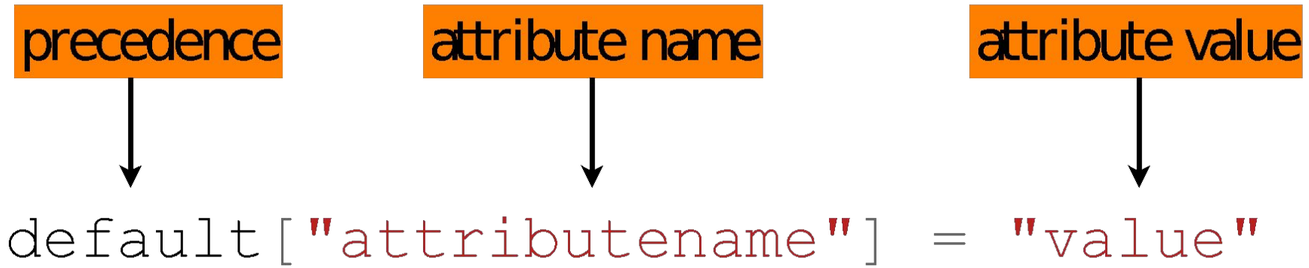
- Well-written cookbooks change behavior based on attributes.
- Ideally, you don't have to modify the contents of a cookbook to use it for your specific use case.
- Look at the attributes directory for things you can override through roles / policyfiles to affect behavior of the cookbook.
- Of course, well written cookbooks have sane defaults, and a README to describe all this.

Setting Attributes in Attribute Files

Cookbook attributes are set in the attributes file

./cookbooks/<cookbook>/attributes/default.rb

Format is:



We'll look at precedence later.

CHEF

Example: Setting package name to an attribute

cookbooks/apache/attributes/default.rb

```
default['apache']['package_name'] = 'httpd'
```

cookbooks/apache/recipes/default.rb

```
package node['apache']['package_name'] do
  action :install
end
```

We can set the name of a particular package to an attribute and then call that attribute within a recipe

Example: Setting package name to an attribute

cookbooks/apache/attributes/default.rb

```
case node['platform']
when 'ubuntu'
 default['apache']['package_name'] = 'apache2'
else
 default['apache']['package_name'] = 'httpd'
end
```

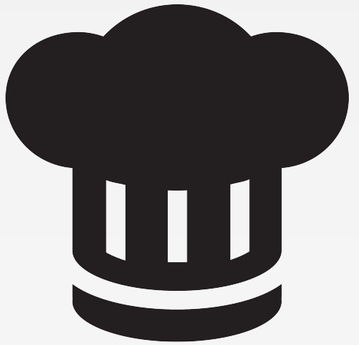
Implementing conditional statements allows us to alter the control flow permitting our cookbooks to be data driven.

Reconfigure Welcome Message

Currently a welcome message is hard coded in both web server cookbooks.

What if we wanted to display a message that includes our company name utilizing a node attribute?

How could we implement this node attribute within both our ‘myiis’ and ‘apache’ cookbooks?



GL: Reconfigure Welcome Message

So we want both our web server cookbooks to display our company name…

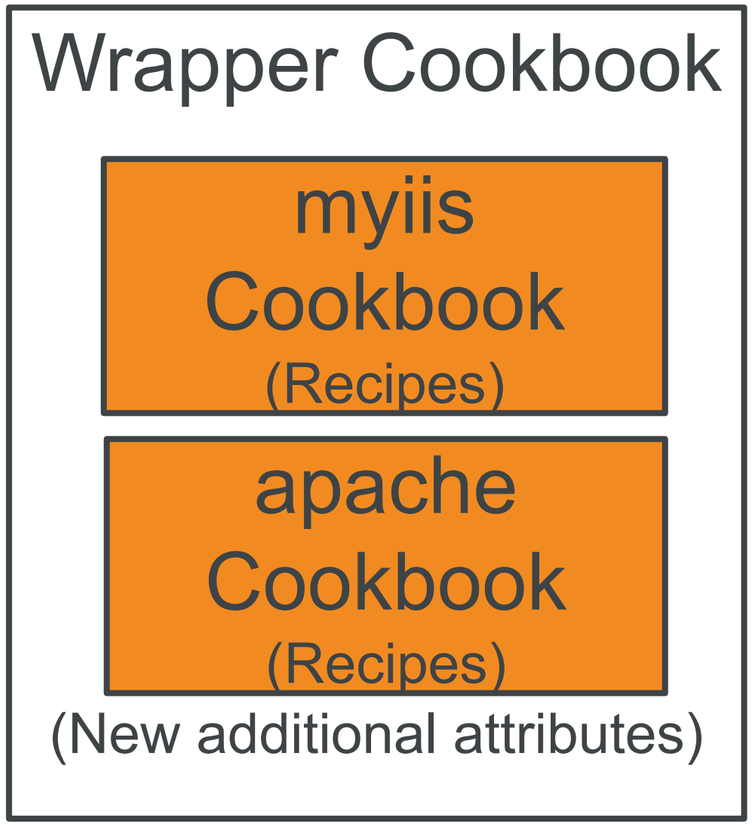
Objective:

- ❑ Create a ‘company_web’ wrapper cookbook that can apply either the ‘myiis’ or ‘apache’ default recipe based on platform
- ❑ Create a node attribute that contains your company name
- ❑ Implement the edit_resource method to update the template resource for both the ‘myiis’ and ‘apache’ server recipes
- ❑ Create Policyfile and lock.
- ❑ Upload Policyfile.lock to the Chef Infra Server
- ❑ Converge the nodes

Wrapper Cookbooks

A wrapper cookbook is a new cookbook that encapsulates the functionality of the original cookbook(s).

It can access all of the recipes, cookbook components, and attributes found in the original cookbook(s) and implement them in new ways.



https://docs.chef.io/supermarket.html#wrapper-cookbooks

https://www.chef.io/blog/2013/12/03/doing-wrapper-cookbooks-right/

GL: Generate the Wrapper Cookbook

```
$ cd ~/chef-repo
$ chef generate cookbook cookbooks/company_web
```

```
Generating cookbook company_web
- Ensuring correct cookbook content

Your cookbook is ready. To setup the pipeline, type `cd
cookbooks/company_web`, then run `delivery init`
```

Do this on your laptop

CHEF

GL: Create Dependency on apache and myiis

~/chef-repo/cookbooks/company_web/metadata.rb

```
name 'company_web'
maintainer 'The Authors'
maintainer_email 'you@example.com'
license 'All Rights Reserved'
description 'Installs/Configures company_web'
long_description 'Installs/Configures company_web'
version '0.1.0'
chef_version '>= 15.0'
depends 'myiis'
depends 'apache'
```


CHEF

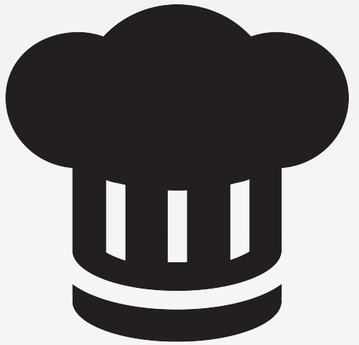
GL: Include Recipe Based on Platform

~/chef-repo/cookbooks/company_web/recipes/default.rb

```
#
# Cookbook:: company_web
# Recipe:: default
#
# Copyright:: 2021, The Authors, All Rights Reserved.

case node['platform']
when 'windows'
  include_recipe 'myiis::default'
else
  include_recipe 'apache::default'
end
```


CHEF



GL: Reconfigure Welcome Message

So we want both our web server cookbooks to display our company name…

Objective:

- ✔ Create a ‘company_web’ wrapper cookbook that can apply either the ‘myiis’ or ‘apache’ default recipe based on platform
- ❑ Create a node attribute that contains your company name
- ❑ Implement the edit_resource method to update the template resource for both the ‘myiis’ and ‘apache’ server recipes
- ❑ Upload cookbooks to the Chef Infra Server via a policyfile
- ❑ Update the run list of the iis_web node to use the default recipe of the company_web cookbook and converge the node

GL: Generate the default Attribute File

```
$ cd ~/chef-repo
$ chef generate --help
```

```
Available generators:
  app              Generate an application repo
  cookbook         Generate a single cookbook
  recipe           Generate a new recipe
  attribute        Generate an attributes file
  template         Generate a file template
  file             Generate a cookbook file
```


CHEF

GL: Generate the default Attribute File

```
$ chef generate attribute --help
```

```
Usage: chef generate attribute [path/to/cookbook] NAME [options]
    -C, --copyright COPYRIGHT        Name of the copyright holder - defaults
to 'The Authors'
    -m, --email EMAIL                Email address of the author - defaults
to 'you@example.com'
    -a, --generator-arg KEY=VALUE    Use to set arbitrary attribute KEY to
VALUE in the code_generator cookbook
    -h, --help                       Show this message
```


CHEF

GL: Generate the default Attribute File

```
$ chef generate attribute cookbooks/company_web default
```

```
Recipe: code_generator::attribute
  * directory[cookbooks/company_web/attributes] action create
    - create new directory cookbooks/company_web/attributes
  * template[cookbooks/company_web/attributes/default.rb] action
create
    - create new file cookbooks/company_web/attributes/default.rb
    - update content in file
cookbooks/company_web/attributes/default.rb from none to e3b0c4
```


CHEF

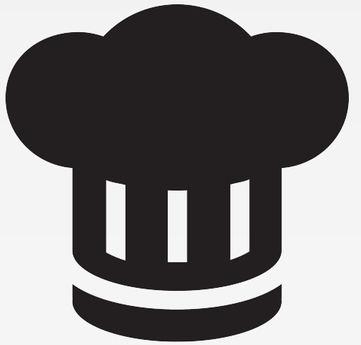
GL: Set the Company Name as an Attribute

`cookbooks/company_web/attributes/default.rb`

```
default['company_web']['company_name'] = 'Your Company Name'
```

You can replace the `Your Company Name` value with **Chef** or any name you like.

CHEF



GL: Reconfigure Welcome Message

So we want both our web server cookbooks to display our company name…

Objective:

- ✔ Create a ‘company_web’ wrapper cookbook that can apply either the ‘myiis’ or ‘apache’ default recipe based on platform
- ✔ Create a node attribute that contains your company name
- ❑ Implement the edit_resource method to update the template resource for both the ‘myiis’ and ‘apache’ server recipes
- ❑ Create Policyfile and lock.
- ❑ Upload Policyfile.lock to the Chef Infra Server
- ❑ Converge the nodes

Using the company_name Attribute

We are now able to apply a different default recipe based on whether the node's platform is Windows or Centos, but how do we update the respective template file to display the company_name attribute for both the 'myiis' and 'apache' cookbooks?

edit_resource

A recipe can find a resource in the resource collection, and then edit it by using the edit_resource method. If a resource block with the same name exists in the resource collection, it will be updated with the contents of the resource block.

https://docs.chef.io/dsl_recipe.html#edit-resource

GL: View the server Recipes

/myiis/recipes/server.rb

```
powershell_script 'Install IIS' do
  code 'Add-WindowsFeature Web-Server'
end

template 'c:\inetpub\wwwroot\Default.htm' do
  source 'Default.htm.erb'
end

service 'w3svc' do
  action [:enable, :start]
end
```

/apache/recipes/server.rb

```
package 'httpd'

template '/var/www/html/index.html' do
  source 'index.html.erb'
end

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

We want to use a new source for the template resource for both our cookbooks

CHEF

GL: Edit the Template resource for myiis

~/chef-repo/cookbooks/company_web/recipes/default.rb

```
case node['platform']
when 'windows'
  include_recipe 'myiis::default'

  edit_resource(:template, 'c:\inetpub\wwwroot\Default.htm') do
    source 'homepage.erb'
    cookbook 'company_web'
  end
#else statement...
```

GL: Edit the Template resource for apache

~/chef-repo/cookbooks/company_web/recipes/default.rb

```
#When Statement...
else
  include_recipe 'apache::default'

  edit_resource(:template, '/var/www/html/index.html') do
    source 'homepage.erb'
    cookbook 'company_web'
  end
end
```

GL: View the default Recipe

~/chef-repo/cookbooks/company_web/recipes/default.rb

```
case node['platform']
when 'windows'
  include_recipe 'myiis::default'

  edit_resource(:template, 'c:\inetpub\wwwroot\Default.htm') do
    source 'homepage.erb'
    cookbook 'company_web'
  end

else
  include_recipe 'apache::default'
  edit_resource(:template, '/var/www/html/index.html') do
    source 'homepage.erb'
    cookbook 'company_web'
  end
end
```

GL: Generate the Template file

```
$ chef generate template cookbooks/company_web homepage
```

```
Recipe: code_generator::template
  * directory[cookbooks/company_web/templates] action create
    - create new directory cookbooks/company_web/templates
  * template[cookbooks/company_web/templates/homepage.erb] action create
    - create new file cookbooks/company_web/templates/homepage.erb
    - update content in file cookbooks/company_web/templates/homepage.erb
from none to e3b0c4
    (diff output suppressed by config)
```


CHEF

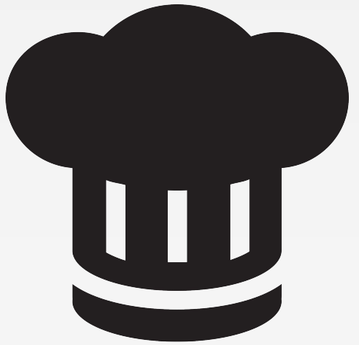
GL: Update the Template File

~/chef-repo/cookbooks/company_web/templates/homepage.erb

```
<html>
  <body>
    <h1><%= node['company_web']['company_name'] %> Welcomes You!</h1>
    <h2>PLATFORM: <%= node['platform'] %></h2>
    <h2>HOSTNAME: <%= node['hostname'] %></h2>
    <h2>MEMORY:   <%= node['memory']['total'] %></h2>
    <h2>CPU Mhz:  <%= node['cpu']['0']['mhz'] %></h2>
  </body>
</html>
```

Note: We are adding all this code but we are also highlighting the `['company_web']['company_name']` attributes for the discussion below.

CHEF



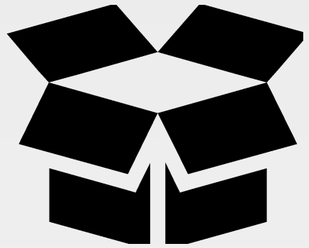
GL: Reconfigure Welcome Message

So we want both our web server cookbooks to display our company name…

Objective:

- ✔ Create a ‘company_web’ wrapper cookbook that can apply either the ‘myiis’ or ‘apache’ default recipe based on platform
- ✔ Create a node attribute that contains your company name
- ✔ Implement the edit_resource method to update the template resource for both the ‘myiis’ and ‘apache’ server recipes
- ❑ Create Policyfile and lock.
- ❑ Upload Policyfile.lock to the Chef Infra Server
- ❑ Converge the nodes

CONCEPT



Policyfile.rb and the Policyfile.lock.json

Now that we have our company_web cookbook in our chef-repo, we can create our Policyfile.rb and then generate our Policyfile.lock.json as we discussed in the previous module.

We'll name our Policyfile **company_web**.

GL: Generate the Policyfile and Name it company_web

```
> cd ~/chef-repo
> chef generate policyfile policyfiles/company_web
```

```
* template[/Users/sdelfante/chef-repo/company_web.rb] action create
    - create new file /Users/sdelfante/chef-repo/company_web.rb
    - update content in file /Users/sdelfante/chef-repo/company_web.rb from
none to 32a368
```

GL: Verify that the Policyfile Exists

```
> ls policyfiles (or dir policyfiles for Windows)
```

```
company_web.rb
```


CHEF

GL: Edit the New company_web.rb Policyfile

~/chef-repo/policyfiles/company_web.rb

```
#...skipping for brevity…
# https://docs.chef.io/policyfile.html
# A name that describes what the system you're building with Chef does.
name 'company_web'

# Where to find external cookbooks:
default_source :supermarket

# run_list: chef-client will run these recipes in the order specified.
run_list 'company_web::default'

# Specify a custom source for a single cookbook:
cookbook 'company_web', path: '../cookbooks/company_web'
cookbook 'myiis', path: '../cookbooks/myiis'
cookbook 'apache', path: '../cookbooks/apache'
```

Add the code in green.

CHEF

GL: Generate the company_web.lock.json

```
~/chef-repo> chef install policyfiles/company_web.rb
```

```
Expanded run list: recipe[company_web::default]
Caching Cookbooks...
Installing company_web >= 0.0.0 from path
Installing myiis       >= 0.0.0 from path
Installing apache      >= 0.0.0 from path


Lockfile written to
/Users/sdelfante/chef-repo/policyfiles/company_web.lock.json
Policy revision id:
3a188723402907081953347 0d82616674e38b200413d82296af7383602b2bf0d
```


CHEF

GL: Verify that the company_web.lock.json Exists

```
> ls policyfiles(or dir for Windows)
```

```
company_web.lock.json  company_web.rb
```


CHEF

GL: Move to the policyfiles directory

```
~/chef-repo> cd policyfiles
```


CHEF

GL: Push the company_web.lock.json to Chef Infra Server

```
~/chef-repo/policyfiles> chef push prod
company_web.lock.json
```

```
ploading policy company_web (3a18872340) to policy group prod
Uploaded apache      0.1.0 (1388ab3a)
Uploaded company_web 0.1.0 (b9969c7c)
Uploaded myiis       0.2.1 (cd0db3ed)
```


CHEF

GL: Verify the Policyfile.lock.json is on Chef Infra Server

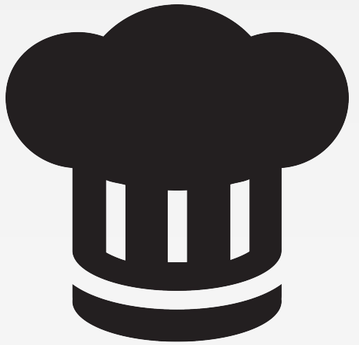
```
~/chef-repo> chef show-policy
```

```
company_web
===========

* prod:  3a18872340
```

Here we can see that the **company_web** policy has been uploaded to Chef Infra Server and is in the **prod** policy group.

CHEF



GL: Reconfigure Welcome Message

So we want both our web server cookbooks to display our company name…

Objective:

- ✔ Create a ‘company_web’ wrapper cookbook that can apply either the ‘myiis’ or ‘apache’ default recipe based on platform
- ✔ Create a node attribute that contains your company name
- ✔ Implement the edit_resource method to update the template resource for both the ‘myiis’ and ‘apache’ server recipes
- ✔ Create Policyfile and lock.
- ✔ Upload Policyfile.lock to the Chef Infra Server
- ❑ Converge the nodes

GL: Apply the company_web Policy to Your Windows Node

```
> knife node policy set iis_web prod company_web
Successfully set the policy on node iis_web
```

node name

policy_group

policy_name

‘knife node policy set` takes 3 arguments:
* node name
* policy group
* policy name

CHEF

GL: View Information About Your Windows Node

```
$ knife node show iis_web
```

```
Node Name:   iis_web
Policy Name:  company_web
Policy Group: prod
FQDN:        WIN-DQFQCUFHDCP.ec2.internal
IP:          3.88.178.251
Run List:
Recipes:
Platform:    windows 6.3.9600
Tags:
```


CHEF

GL: Apply the company_web Policy to Your Linux Node

```
> knife node policy set apache_web prod company_web
Successfully set the policy on node apache_web
```

node name

policy_group

policy_name

‘knife node policy set` takes 3 arguments:
* node name
* policy group
* policy name

CHEF

GL: View Information About Your Linux Node

```
$ knife node show apache_web
```

```
Node Name:   apache_web
Policy Name:  company_web
Policy Group: prod
FQDN:        ip-172-31-62-68.ec2.internal
IP:          18.206.64.141
Run List:
Recipes:
Platform:    centos 7.6.1810
Tags:
```

GL: Converge your Windows Node via winrm

```
$ knife winrm IPADDRESS -m -x Administrator -P PASSWORD "chef-client"
```

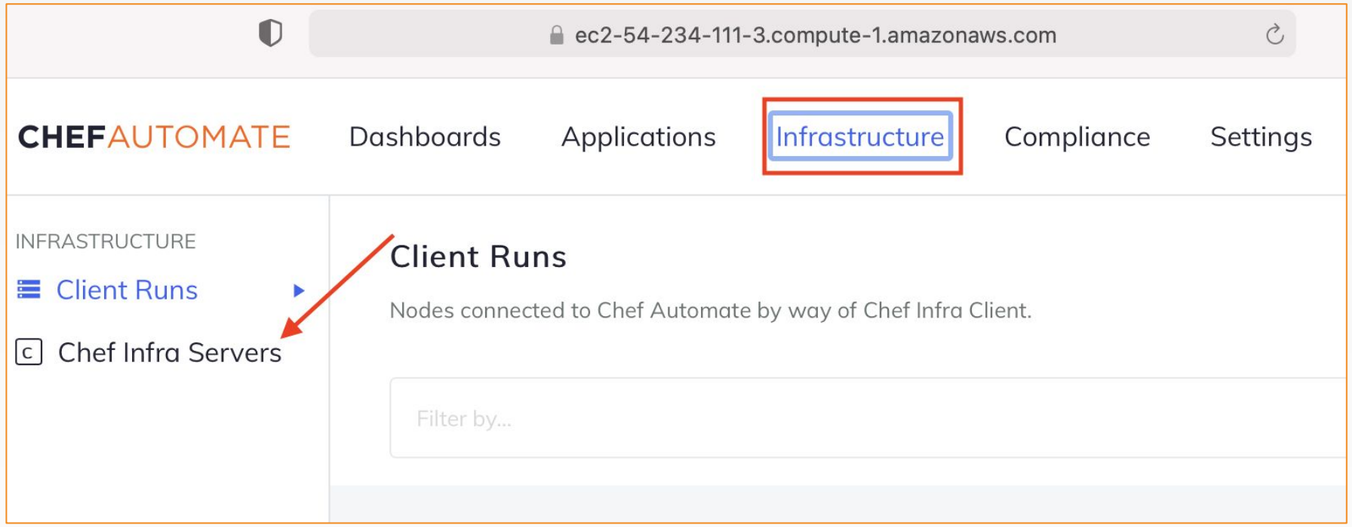
```
…
Synchronizing Cookbooks:
- myiis (0.2.1)
3.88.178.251   - apache (0.1.0)
3.88.178.251
3.88.178.251 - company_web (0.1.0)
3.88.178.251 Installing Cookbook Gems:
3.88.178.251 Compiling Cookbooks...
* windows_service[w3svc] action start (up to date)
...
3.88.178.251 Running handlers:
3.88.178.251 Running handlers complete
3.88.178.251 Chef Infra Client finished, 2/4 resources updated in 48 seconds
```

Verify policy applied to iis_web

1. Sign In to Chef Automate server.
2. Click on Infrastructure tab and Go to Chef Infra Servers (Under Left Navigation Panel)
3. Click on Listed Chef Infra Server - 'cheftraining'
4. Click on the organization - 'studentxx'
5. Click on Nodes tab and select node iis_web.
6. View Information associated with node.

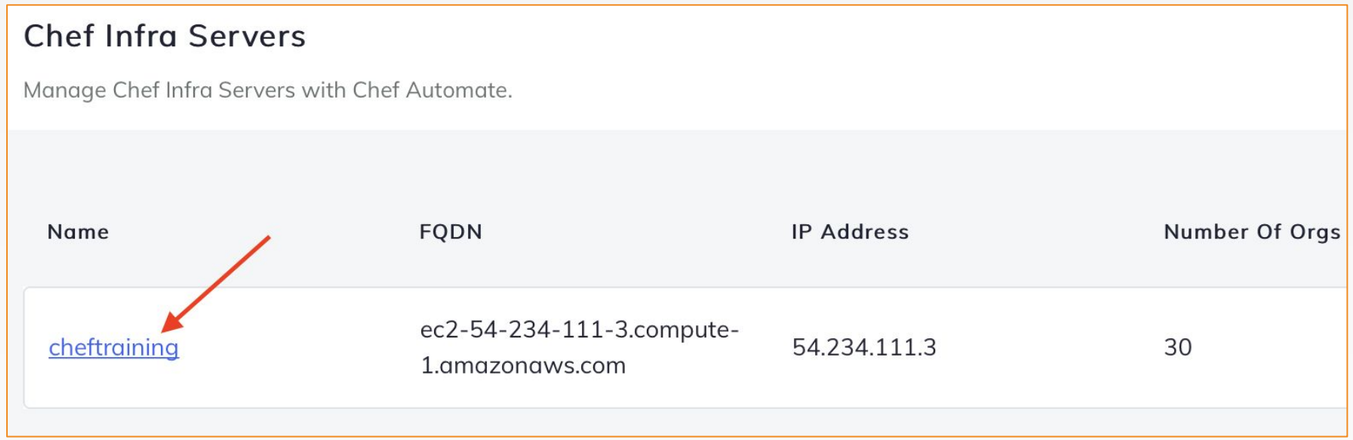
Verify policy applied to iis_web

1. Sign In to Chef Automate server.
2. Click on Infrastructure tab and Go to Chef Infra Servers (Under Left Navigation Panel)
3. Click on Listed Chef Infra Server - 'cheftraining'
4. Click on the organization - 'studentxx'
5. Click on Nodes tab and select node iis_web.
6. View Information associated with node.



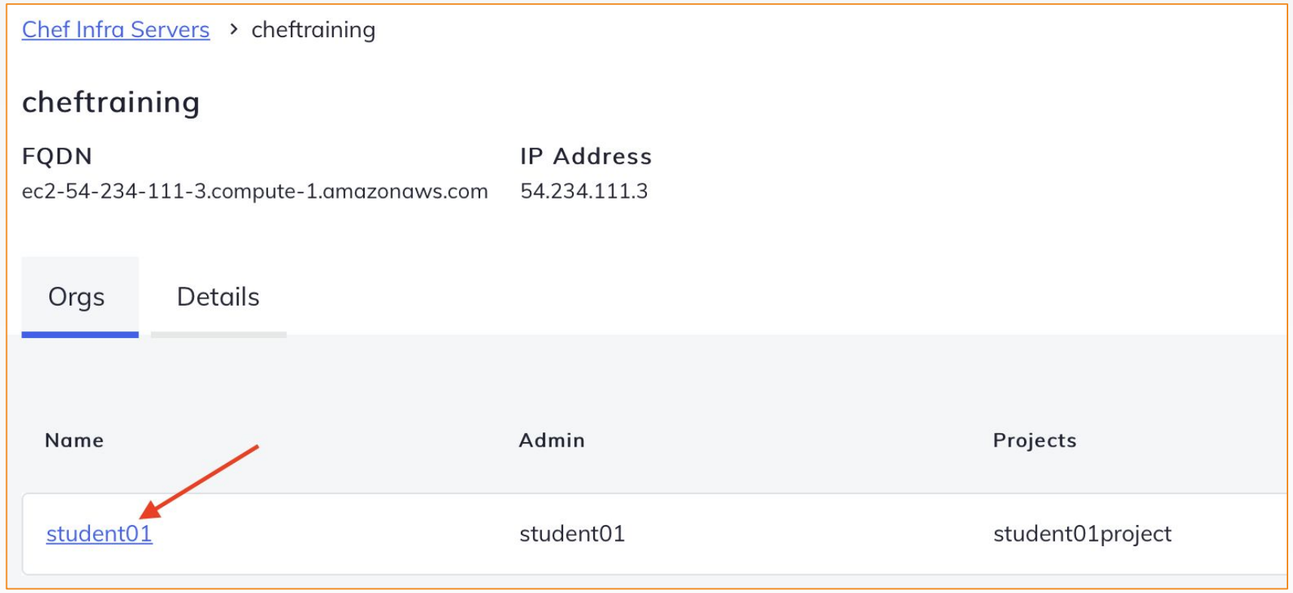
Verify policy applied to iis_web

1. Sign In to Chef Automate server.
2. Click on Infrastructure tab and Go to Chef Infra Servers (Under Left Navigation Panel)
3. Click on Listed Chef Infra Server - 'cheftraining'
4. Click on the organization - 'studentxx'
5. Click on Nodes tab and select node iis_web.
6. View Information associated with node.



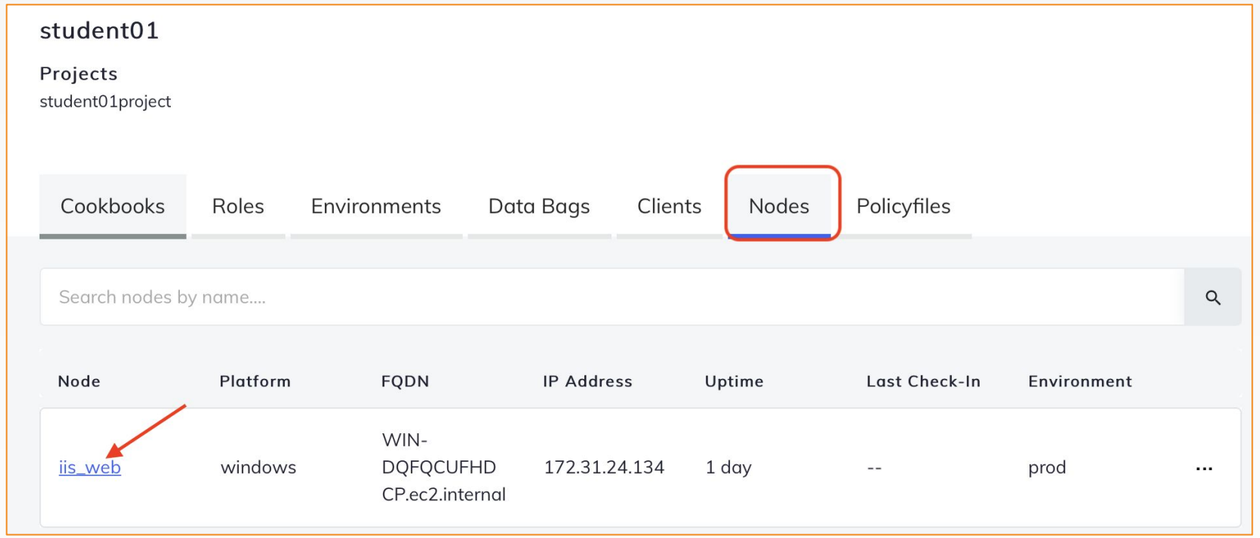
Verify policy applied to iis_web

1. Sign In to Chef Automate server.
2. Click on Infrastructure tab and go to Chef Infra Servers (Under Left Navigation Panel)
3. Click on Listed Chef Infra Server - 'cheftraining'
4. Click on the organization - 'studentxx'.
5. Click on Nodes tab and select node iis_web.
6. View Information associated with node.



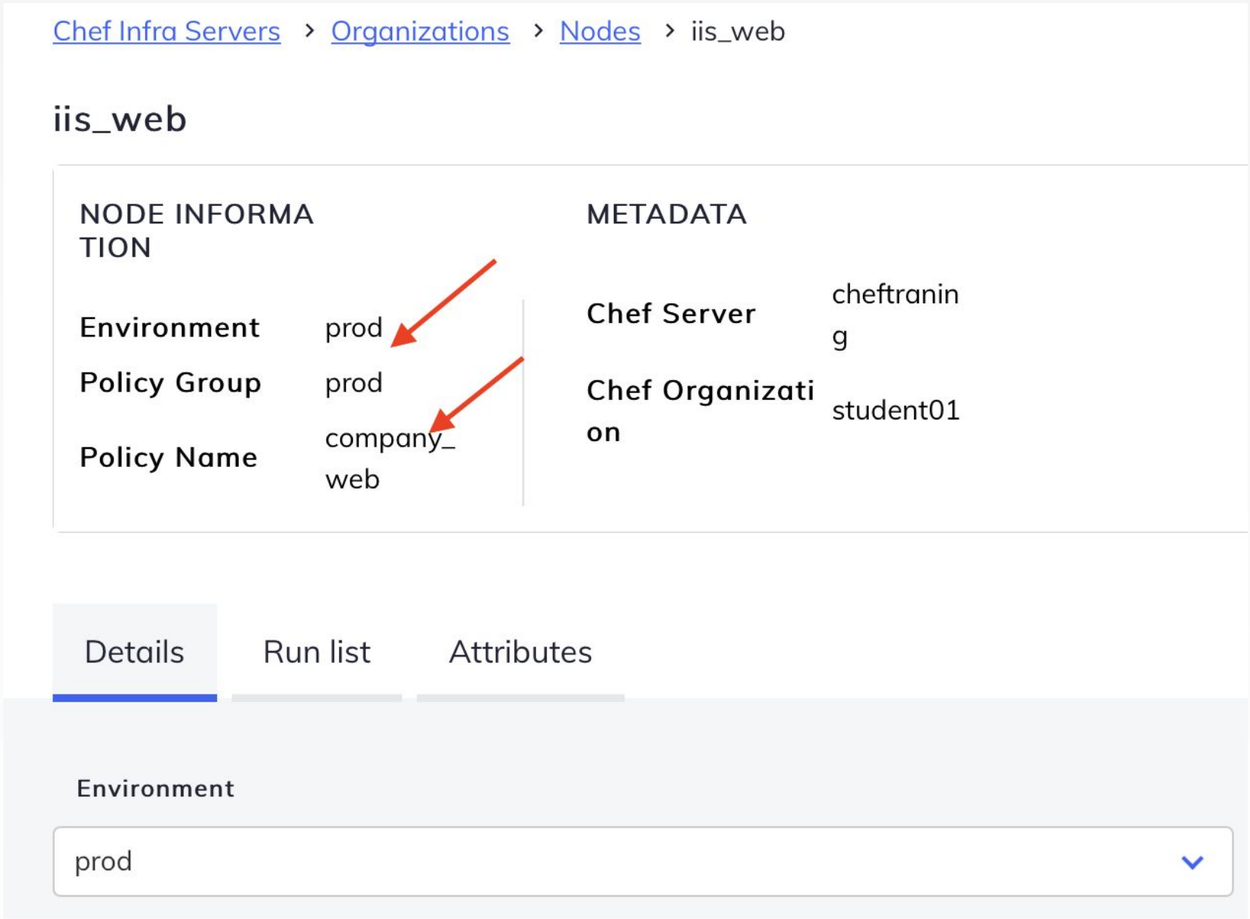
Verify policy applied to iis_web

1. Sign In to Chef Automate server.
2. Click on Infrastructure tab and go to Chef Infra Servers (Under Left Navigation Panel)
3. Click on Listed Chef Infra Server - 'cheftraining'
4. Click on the organization - 'studentxx'.
5. Click on Nodes tab and select node iis_web.
6. View Information associated with node.



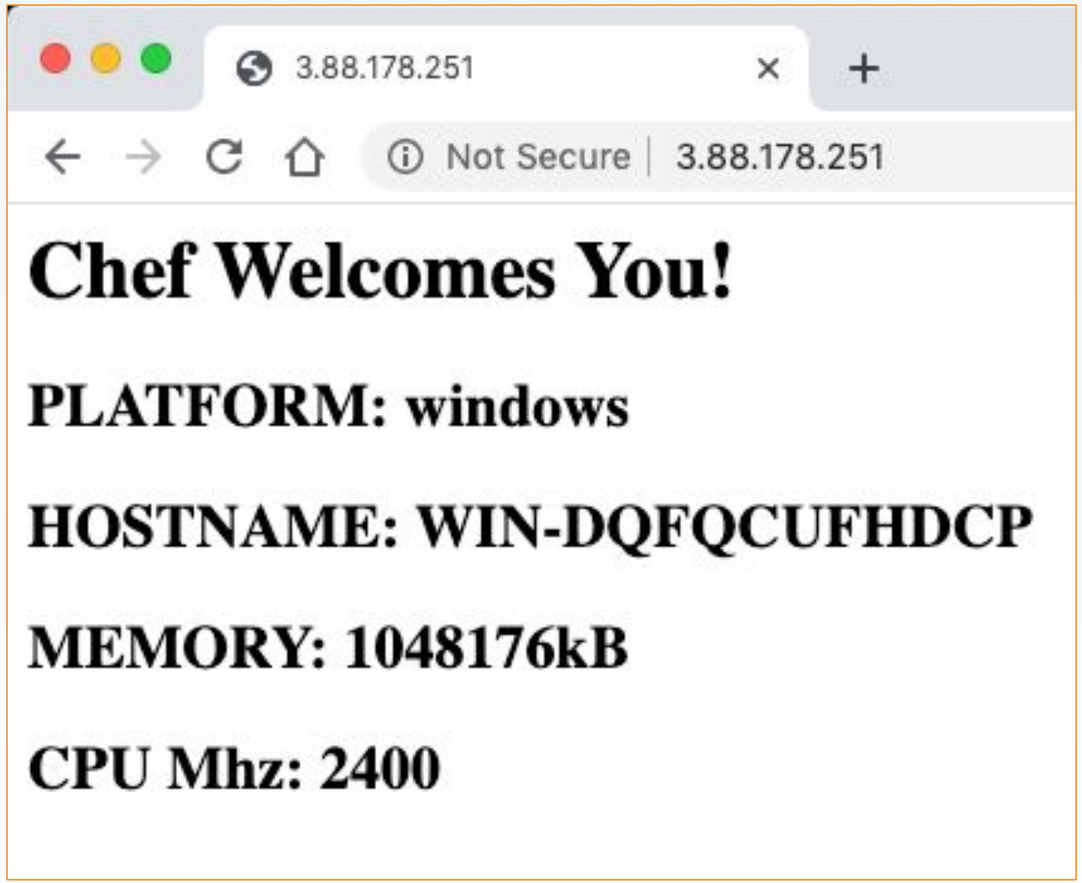
Verify policy applied to iis_web

1. Sign In to Chef Automate server.
2. Click on Infrastructure tab and go to Chef Infra Servers (Under Left Navigation Panel)
3. Click on Listed Chef Infra Server - 'cheftraining'
4. Click on the organization - 'studentxx'.
5. Click on Nodes tab and select node iis_web.
6. View Information associated with node.



CHEF

GL: Verify that the Windows Node Serves the Page



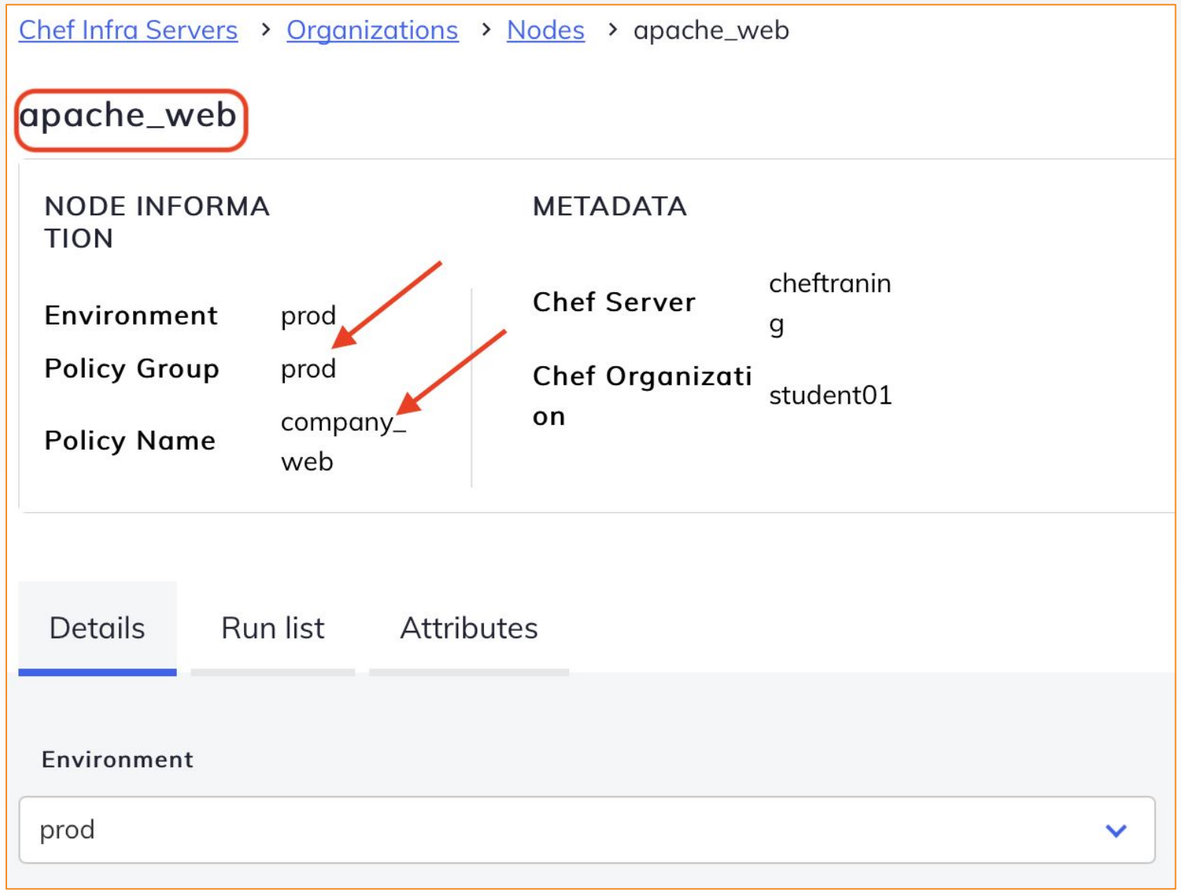
GL: Converge the Linux Node with ssh

```
$ knife ssh IPADDRESS -m -x chef -P PWD 'sudo chef-client'
```

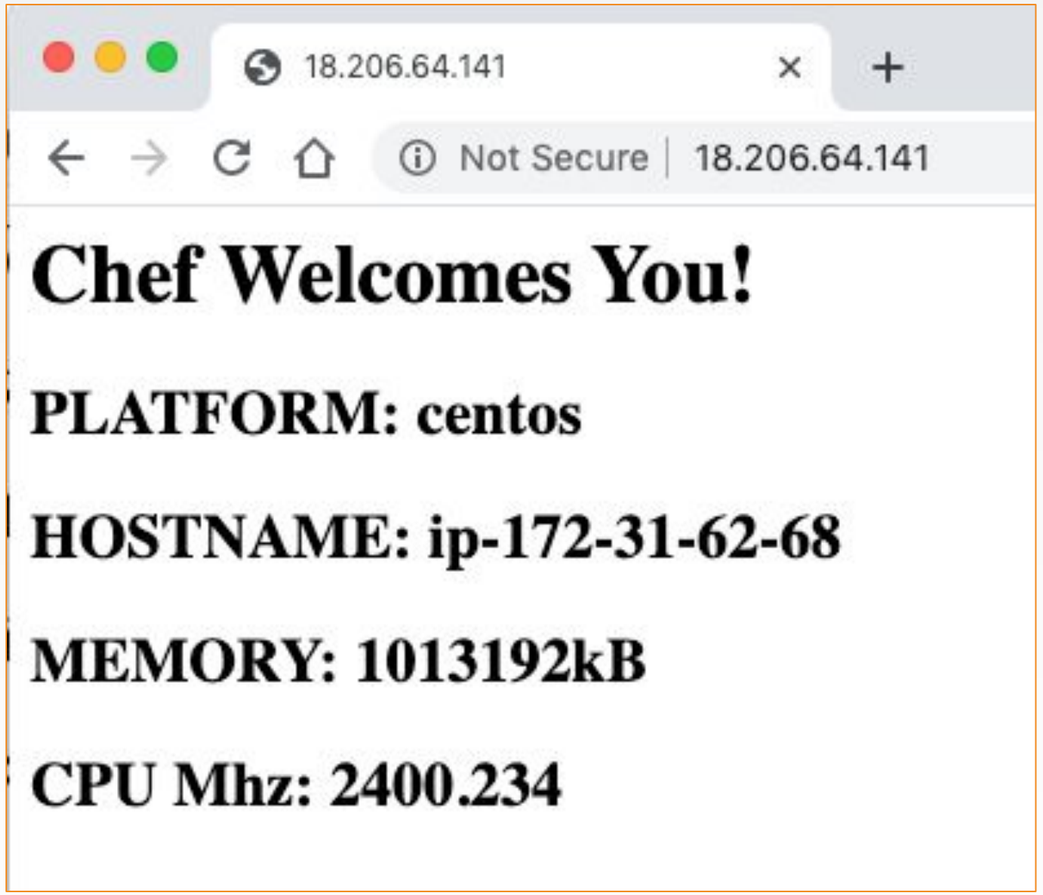
```
…
Synchronizing Cookbooks:
- myiis (0.2.1)
3.88.178.251   - apache (0.1.0)
3.88.178.251
3.88.178.251 - company_web (0.1.0)
3.88.178.251 Installing Cookbook Gems:
3.88.178.251 Compiling Cookbooks...
* windows_service[w3svc] action start (up to date)
...
3.88.178.251 Running handlers:
3.88.178.251 Running handlers complete
3.88.178.251 Chef Infra Client finished, 2/4 resources updated in 48 seconds
```

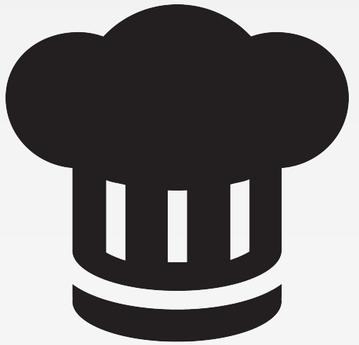
Verify policy applied to apache_web

View Information associated with node.



GL: Verify that the Linux Node Serves the Page





GL: Reconfigure Welcome Message

So we want both our web server cookbooks to display our company name…

Objective:

- ✔ Create a ‘company_web’ wrapper cookbook that can apply either the ‘myiis’ or ‘apache’ default recipe based on platform
- ✔ Create a node attribute that contains your company name
- ✔ Implement the edit_resource method to update the template resource for both the ‘myiis’ and ‘apache’ server recipes
- ✔ Create Policyfile and lock.
- ✔ Upload Policyfile.lock to the Chef Infra Server
- ✔ Converge the node

Q&A

What questions can we answer for you?

CHEF

CHEF™