

# Running chef-client as a Service

Introducing the chef-client cookbook

# Lesson Objectives

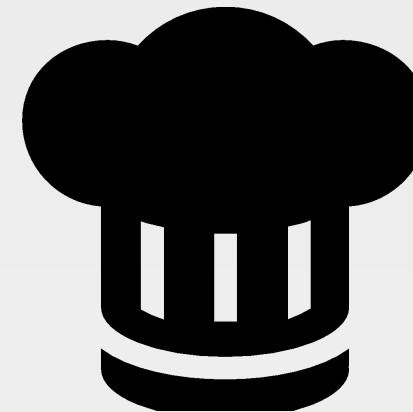
After completing this module, you should be able to:

- ❑ Generate a chef-client wrapper cookbook
- ❑ Update the company\_web policyfile to include the chef-client wrapper cookbook
- ❑ Update and push the company\_web Policyfile to Chef Infra Server
- ❑ Apply the updated Policyfile to our web nodes
- ❑ Run chef-client as a service/task on our web nodes



# Step Back: How is chef-client Configured?

- ❖ How can I run chef-client as a service or Windows task?
- ❖ Where can I configure logging?
- ❖ How does chef-client know what Chef Server to connect to?
- ❖ How does chef-client authenticate with the Chef Server?
- ❖ How do I configure where chef-client caches?



# GL: View How chef-client is Configured

In this group lab you will view how chef-client is configured.

# Demo: View chef-client config Directory (Linux)



```
$ knife ssh "os:linux" -x USER -P PWD "ls -F /etc/chef"
```

```
ec2-34-196-104-17.compute-1.amazonaws.com accepted_licenses/ chef_guid client.pem  
client.rb first-boot.json ohai/  
ec2-34-196-50-77.compute-1.amazonaws.com accepted_licenses/ chef_guid client.pem  
client.rb first-boot.json ohai/
```

# Demo: View chef-client config File (Linux)



```
$ knife ssh "os:linux" -x USER -P PWD "cat  
/etc/chef/client.rb"
```

```
ec2-54-209-220-6.compute-1.amazonaws.com chef_server_url "https://api.chef.io/organizations/jane3"  
ec2-54-209-220-6.compute-1.amazonaws.com validation_client_name "jane3-validator"  
ec2-54-209-220-6.compute-1.amazonaws.com chef_license "accept"  
ec2-54-209-220-6.compute-1.amazonaws.com log_location STDOUT  
ec2-54-209-220-6.compute-1.amazonaws.com node_name "lb"  
ec2-54-209-220-6.compute-1.amazonaws.com  
ec2-18-206-64-141.compute-1.amazonaws.com chef_server_url "https://api.chef.io/organizations/jane3"  
ec2-18-206-64-141.compute-1.amazonaws.com validation_client_name "jane3-validator"  
ec2-18-206-64-141.compute-1.amazonaws.com chef_license "accept"  
ec2-18-206-64-141.compute-1.amazonaws.com log_location STDOUT  
ec2-18-206-64-141.compute-1.amazonaws.com node_name "apache_web"  
ec2-18-206-64-141.compute-1.amazonaws.com
```

# Demo: View chef-client config Directory (Windows)



```
$ knife winrm "os:windows" -x USER -P PWD  
-a cloud.public_ipv4 "dir C:\chef"
```

```
3.88.178.251 Volume in drive C has no label.  
3.88.178.251 Volume Serial Number is D4E5-7A7A  
3.88.178.251 Directory of C:\chef  
3.88.178.251 02/11/2020 07:01 PM <DIR> .  
3.88.178.251 02/11/2020 07:01 PM <DIR> ..  
3.88.178.251 02/11/2020 07:01 PM <DIR> backup  
3.88.178.251 02/11/2020 05:42 PM <DIR> cache  
3.88.178.251 02/10/2020 09:40 PM 36 chef_guid  
3.88.178.251 02/10/2020 09:39 PM 1,706 client.pem  
3.88.178.251 02/10/2020 09:39 PM 349 client.rb  
3.88.178.251 02/10/2020 09:39 PM 17 first-boot.json  
3.88.178.251 02/01/2016 06:41 AM <DIR> ohai
```

# Demo: View chef-client config File (Windows)



```
$ knife winrm "os:windows" -x USER -P PWD  
-a cloud.public_ipv4 "more C:\chef\client.rb"
```

```
3.88.178.251 chef_server_url "https://api.chef.io/organizations/jane3"  
3.88.178.251  
3.88.178.251 validation_client_name "jane3-validator"  
3.88.178.251 file_cache_path "c:/chef/cache"  
3.88.178.251 file_backup_path "c:/chef/backup"  
3.88.178.251 cache_options ({:path => "c:/chef/cache/checksums", :skip_expires => true})  
3.88.178.251 chef_license "accept"  
3.88.178.251 node_name "iis_web"  
3.88.178.251 log_level :auto  
3.88.178.251 log_location STDOUT
```



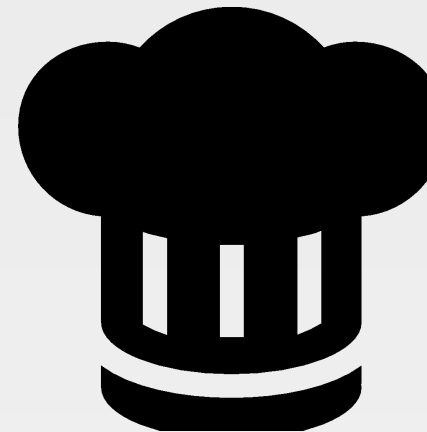


# Introducing the chef-client Cookbook

The chef-client cookbook allows you to manage and configure chef-client as a service on Linux-based nodes, or as a task on Windows nodes, configure logging, caching, etc.

Bootstrapping installs the chef-client executable.

The chef-client cookbook is used to configure chef-client.



# GL: The chef-client Cookbook

*We will use the chef-client community cookbook to configure chef-client on each of our nodes to run periodically*

## Objective:

- ☐ Generate a chef-client wrapper cookbook
- ☐ Update the company\_web policyfile to include the chef-client wrapper cookbook
- ☐ Push the company\_web policyfile to Chef Infra Server
- ☐ Apply the updated Policyfile to our web nodes



# Wrapper Cookbooks

Don't use forked community cookbooks in production, or you will miss out on upstream changes, and will have to rebase

Instead use **wrapper cookbooks** to wrap upstream cookbooks and change their behavior without forking

# GL: Create mychef\_client Wrapper Cookbook



Generating cookbook mychef\_client

- Ensuring correct cookbook content
- Committing cookbook files to git

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

# GL: Update mychef\_client metadata.rb

`~/chef-repo/cookbooks/mychef_client/metadata.rb`

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

depends 'chef-client'
```

# GL: Edit mychef\_client Default Recipe



```
cookbooks/mychef_client/recipes/default.rb
```

```
#  
# Cookbook:: mychef_client  
# Recipe:: default  
#  
# Copyright:: 2020, The Authors, All Rights Reserved.  
  
include_recipe 'chef-client::default'
```

This recipe just calls the recipe `chef-client::default`

# GL: Edit company\_web.rb Policyfile

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

```
...# run_list: chef-client will run these recipes in the order
specified.
run_list 'mychef_client::default', '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'
cookbook 'mychef_client', path: '../cookbooks/mychef_client'
```

Add the **mychef\_client** cookbook to the run list and cookbook paths.

# GL: Update the Policyfile



```
$ chef update policyfiles/company_web.rb
```

```
...  
Expanded run list: recipe[mychef_client::default],  
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  
Installing mychef_client  >= 0.0.0 from path  
Installing chef-client    12.3.4  
Installing cron           7.0.3  
Installing logrotate      3.0.4  
Lockfile written to  
/Users/sdelfante/chef-repo/policyfiles/company_web.lock.json ...
```



# GL: Push the Policyfile



```
$ chef push prod policyfiles/company_web.lock.json
```

```
Uploading policy company_web (fd0a5ffe1c) to policy group prod
```

```
Using      apache      0.1.0    (1388ab3a)
```

```
Using      company_web  0.1.1    (085c5742)
```

```
Using      myiis        0.2.1    (c7630da4)
```

```
Uploaded chef-client    12.3.4    (7cb128f1)
```

```
Uploaded cron           7.0.3     (602e43b3)
```

```
Uploaded logrotate      3.0.4     (bd20a5c5)
```

```
Uploaded mychef_client  0.1.0     (c7c59f9a)
```

# GL: Converge the Linux Web Node



```
$ knife ssh 'name:apache_web' -x chef -P PWD 'sudo chef-client'
```

```
ec2-18-206-64-141.compute-1.amazonaws.com Using policy 'company_web' at  
revision 'fd0a5ffe1cf4748f69bc66076de563fa2d2214ca16803d69ad89f7dbfc09fa39'  
ec2-18-206-64-141.compute-1.amazonaws.com resolving cookbooks for run list:  
...  
ec2-18-206-64-141.compute-1.amazonaws.com Synchronizing Cookbooks:  
ec2-18-206-64-141.compute-1.amazonaws.com - apache (0.1.0)  
ec2-18-206-64-141.compute-1.amazonaws.com - company_web (0.1.0)  
ec2-18-206-64-141.compute-1.amazonaws.com - chef-client (12.3.4)  
ec2-18-206-64-141.compute-1.amazonaws.com - myiis (0.2.1)  
ec2-18-206-64-141.compute-1.amazonaws.com - logrotate (3.0.4)  
ec2-18-206-64-141.compute-1.amazonaws.com - cron (7.0.2)  
ec2-18-206-64-141.compute-1.amazonaws.com - mychef_client (0.1.0)
```

# GL: Verify chef-client is Running (Linux)



```
$ knife ssh 'name:apache_web' -x chef -P PWD "ps awux |  
grep chef-client"
```

```
ec2-18-206-64-141.compute-1.amazonaws.com root      6462  0.9  6.1 299884 62692 ?  
Ss   17:12   0:02 /opt/chef-workstation/embedded/bin/ruby --disable-gems  
/usr/bin/chef-client -c /etc/chef/client.rb -i 1800 -s 300  
ec2-18-206-64-141.compute-1.amazonaws.com chef      6530  0.0  0.1 113180  1604 pts/0  
Ss+  17:17   0:00 bash -c ps awux | grep chef-client  
ec2-18-206-64-141.compute-1.amazonaws.com chef      6546  0.0  0.0 112708   972 pts/0  
S+   17:17   0:00 grep chef-client
```

Notice that the interval is set to 1800 seconds.



# View node apache\_web run list

In the **Chef Automate** web interface, view the updated run list for the apache\_web server

‘mychef\_client::default’.

[Chef Infra Servers](#) > [Organizations](#) > [Nodes](#) > apache\_web


apache\_web

NODE INFORMATION		METADATA	
Environment	prod	Chef Server	cheftraning
Policy Group	prod	Chef Organization	student01
Policy Name	company_web		

Details **Run list** Attributes

Run List

[+ Expand All](#) [- Collapse All](#)

	Version	Position
 mychef_client::default	...	0

# GL: Converge the iis\_web Node (Windows)



```
$ knife winrm 'name:iis_web' -x USER -P PWD  
-a cloud.public_ipv4 "chef-client"
```

```
34.195.38.226 Using policy 'company_web' at revision  
'7f50faaaf9eceb35f72358c38a71b86c7fb7df8f49fee245827eb4613d919827'  
34.195.38.226  
34.195.38.226 resolving cookbooks for run list: ["mychef_client::default@0.1.0  
(f5b035d)", "company_web::default@0.1.0 (c1b26cb)"]  
34.195.38.226 Synchronizing Cookbooks:  
34.195.38.226 - apache (0.1.0)  
34.195.38.226 - company_web (0.1.0)  
34.195.38.226 - cron (7.0.3)  
34.195.38.226 - myiis (0.2.1)  
34.195.38.226 - logrotate (3.0.3)  
34.195.38.226 - mychef_client (0.1.0) ...
```

# GL: Verify chef-client is Running (Windows)



```
$ knife winrm 'name:iis_web' -x Administrator -P PWD -a  
cloud.public_ipv4 'schtasks /Query | findstr /i "chef-client"'
```

```
3.88.178.251 chef-client 2/13/2020 5:52:00 PM Ready
```



# View node iis\_web run list

In the **Chef Automate** web interface, view the updated run list for the iis\_web server

‘mychef\_client::default’.

[Chef Infra Servers](#) > [Organizations](#) > [Nodes](#) > iis\_web


**iis\_web**

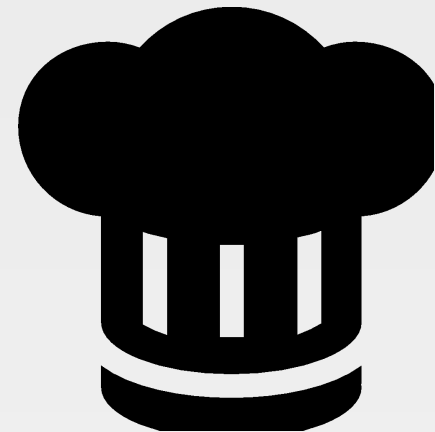
NODE INFORMATION		METADATA	
Environment	prod	Chef Server	cheftraning
Policy Group	prod	Chef Organization	student01
Policy Name	company_web		

Details **Run list** Attributes

Run List

[+ Expand All](#) [- Collapse All](#) [Edit](#)

	Version	Position
 mychef_client::default	...	0



# Introducing chef-client cookbook

*We will use the chef-client community cookbook to configure chef-client on each of our nodes to run periodically*

## Objective:

- ✓ Generate a chef-client wrapper cookbook
- ✓ Update the company\_web policyfile to include the chef-client wrapper cookbook
- ✓ Push the company\_web policyfile to Chef Infra Server
- ✓ Apply the update Policyfile to our web nodes





# GL: Change Default Settings

Wait!

There has just been a mandate that every web node in the infrastructure must run chef-client every 5 minutes (300 seconds).

We'll accomplish this change by creating an attribute file which will override the default setting of 1800 seconds

# GL: Generate the default Attribute File



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

```
Recipe: code_generator::attribute
```

```
  * directory[cookbooks/mychef_client/attributes] action create
```

```
    - create new directory cookbooks/mychef_client/attributes
```

```
  * template[cookbooks/mychef_client/attributes/default.rb] action  
create
```

```
    - create new file
```

```
cookbooks/mychef_client/attributes/default.rb
```

```
    - update content in file
```

```
cookbooks/mychef_client/attributes/default.rb from none to e3b0c4  
(diff output suppressed by config)
```

# GL: Set the chef\_client Interval

`cookbooks/mychef_client/attributes/default.rb`

```
force_default['chef_client']['interval'] = '300'
```

# Node Attribute Precedence

	Attribute Files	Node/Recipe	Policy File
default	1	2	3
force_default	4	5	
normal	6	7	
override	8	9	10
force_override	11	12	
automatic from ohai highest priority			

# GL: Update the Policyfile



```
$ cd ~/chef-repo/policyfiles
$ chef update company_web.rb
```

...

```
Expanded run list: recipe[mychef_client::default],
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
```

```
Installing mychef_client  >= 0.0.0 from path
```

```
Using      chef-client    12.3.4
```

```
Using      cron           7.0.3
```

# GL: Push the Policyfile to Prod



```
$ chef push prod company_web.lock.json
```

```
Uploading policy company_web (1e97a11553) to policy group prod
```

```
Using      apache      0.1.0    (1388ab3a)
```

```
Using      chef-client  12.3.4   (7cb128f1)
```

```
Using      company_web  0.1.0    (085c5742)
```

```
Using      cron         7.0.3    (602e43b3)
```

```
Using      logrotate    3.0.4    (bd20a5c5)
```

```
Using      myiis        0.2.1    (c7630da4)
```

```
Uploaded mychef_client 0.1.0    (f79fa661)
```

# GL: Converge the Linux Web Node



```
$ knife ssh 'name:apache_web' -x chef -P PWD 'sudo chef-client'
```

...

ec2-34-196-104-17.compute-1.amazonaws.com Synchronizing Cookbooks:

ec2-34-196-104-17.compute-1.amazonaws.com - apache (0.1.0)

ec2-34-196-104-17.compute-1.amazonaws.com - chef-client (12.3.4)

ec2-34-196-104-17.compute-1.amazonaws.com - cron (7.0.2)

ec2-34-196-104-17.compute-1.amazonaws.com - myiis (0.2.1)

ec2-34-196-104-17.compute-1.amazonaws.com - logrotate (3.0.3)

ec2-34-196-104-17.compute-1.amazonaws.com - company\_web (0.1.0)

ec2-34-196-104-17.compute-1.amazonaws.com - mychef\_client (0.1.0)

...

\* template[/etc/sysconfig/chef-client] action create

ec2-34-196-104-17.compute-1.amazonaws.com - update content in file  
/etc/sysconfig/chef-client from ec7de1 to 4251d6

ec2-34-196-104-17.compute-1.amazonaws.com --- /etc/sysconfig/chef-client 2020-07-31

16:12:34.501755564 +0000

# GL: Verify chef-client is Running (Linux)



```
$ knife ssh 'name:apache_web' -x chef -P PWD "ps awux |  
grep chef-client"
```

```
ec2-18-206-64-141.compute-1.amazonaws.com root      6864  5.2  6.1 299356 62120 ?  
Ss   17:32   0:02 /opt/chef-workstation/embedded/bin/ruby --disable-gems  
/usr/bin/chef-client -c /etc/chef/client.rb -i 300 -s 300  
ec2-18-206-64-141.compute-1.amazonaws.com chef      6884  0.0  0.1 113180  1600 pts/0  
Ss+  17:32   0:00 bash -c ps awux | grep chef-client  
ec2-18-206-64-141.compute-1.amazonaws.com chef      6900  0.0  0.0 112708   972 pts/0  
S+   17:32   0:00 grep chef-client
```

Notice that the interval is now 300 seconds instead of 1800 as it was before.



# GL: Converge the iis\_web Node (Windows)



```
$ knife winrm 'name:iis_web' -x USER -P PWD  
-a cloud.public_ipv4 "chef-client"
```

```
3.88.178.251 Starting Chef Infra Client, version 17.4.38  
3.88.178.251 Using policy 'company_web' at revision  
'1e97a11553265cefa813fab2898e718040939c5787f9f1a18c3bf2714f82898b'  
3.88.178.251  
3.88.178.251 resolving cookbooks for run list: ["mychef_client::default@0.1.0  
(f79fa66)", "company_web::default@0.1.1 (085c574)"]  
...  
3.88.178.251 * windows_service[w3svc] action enable (up to date)  
3.88.178.251 * 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, 1/10 resources updated in 06 seconds
```

# GL: Verify chef-client is Running (Windows)



```
$ knife winrm 'name:iis_web' -x Administrator -P PWD -a  
cloud.public_ipv4 'schtasks /Query | findstr /i "chef-client"'
```

```
3.88.178.251 chef-client 2/13/2020 5:52:00 PM Ready
```



## Q&A

What questions can we help you answer?

- Chef Supermarket
- Wrapper Cookbooks
- chef-client Cookbook



**CHEF**™