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?

Demo: View chef-client config Directory (Linux)

```
$ knife ssh "os:linux" -x centos -i <aws.pem path> "ls
-F /etc/chef"
```

```
ec2-18-220-243-173.us-east-2.compute.amazonaws.com accepted_licenses/  chef_guid
client.pem  client.rb  first-boot.json


ec2-52-15-221-52.us-east-2.compute.amazonaws.com   accepted_licenses/  chef_guid
client.pem  client.rb  first-boot.json


ec2-3-16-131-207.us-east-2.compute.amazonaws.com   accepted_licenses/  chef_guid
client.pem  client.rb  first-boot.json
```

Demo: View chef-client config File (Linux)

```
$ knife ssh "os:linux" -x centos -i aws.pem "cat
/etc/chef/client.rb"
```

```
ec2-52-15-221-52.us-east-2.compute.amazonaws.com   chef_server_url
"https://api.chef.io/organizations/aspechef23"
ec2-52-15-221-52.us-east-2.compute.amazonaws.com   validation_client_name
"aspechef23-validator"
ec2-52-15-221-52.us-east-2.compute.amazonaws.com   chef_license "accept"
ec2-52-15-221-52.us-east-2.compute.amazonaws.com   log_location   STDOUT
ec2-52-15-221-52.us-east-2.compute.amazonaws.com   node_name "node3"
```

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 update 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

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

```
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 '>= 12.1' if respond_to?(:chef_version)

depends 'chef-client'
```

GL: Edit mychef_client Default Recipe

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

```
#
# Cookbook:: mychef_client
# Recipe:: default
#
# Copyright:: 2019, 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/company_web.rb
```

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

Add the **mychef_client** cookbook to the run list and cookbook paths.

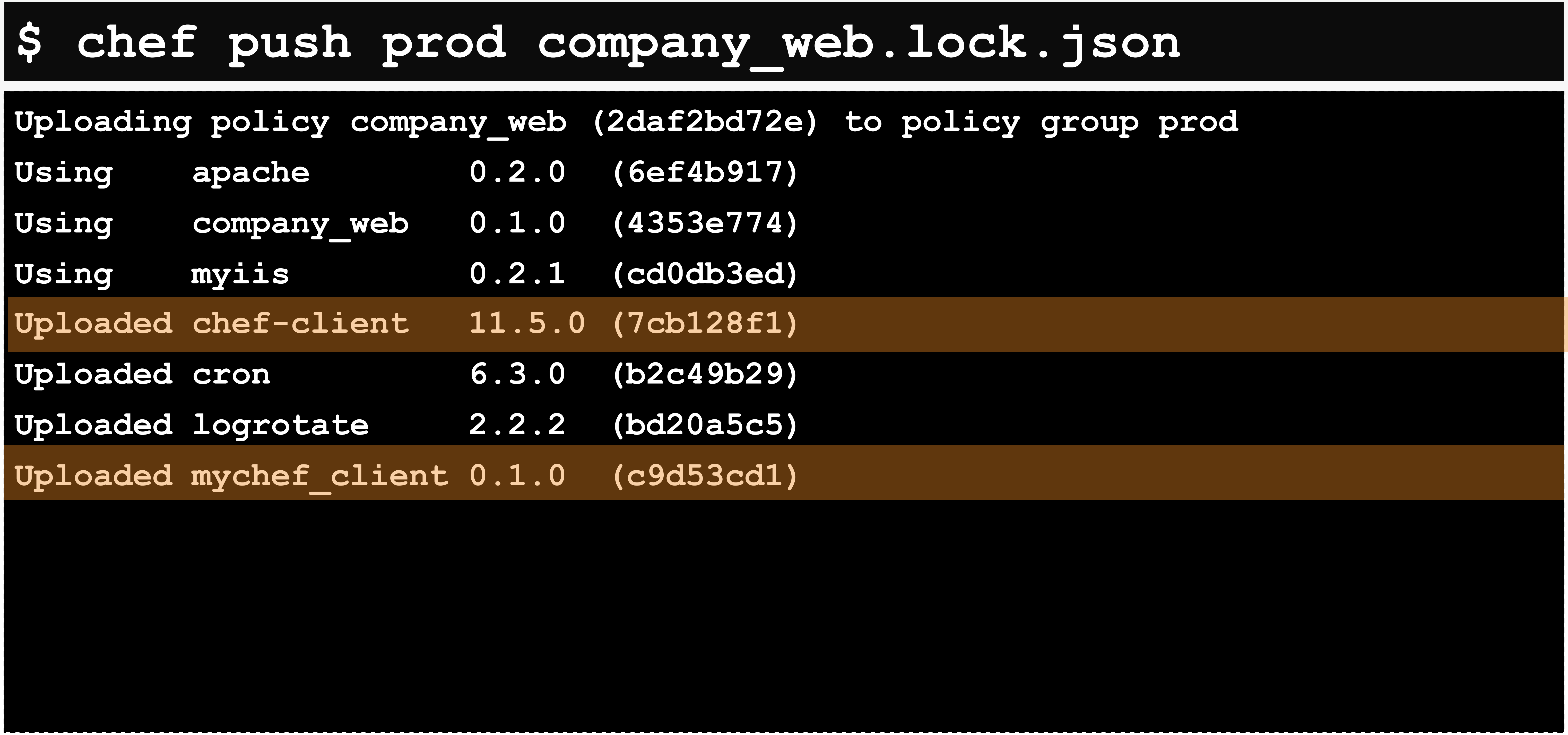
GL: Update the Policyfile

```
$ chef update company_web.rb
```

```
Attributes already up to date
Building policy company_web
Expanded run list: recipe[mychef_client::default], recipe[company_web::default]
Caching Cookbooks...
Installing company_web   >= 0.0.0 from path
Installing apache        >= 0.0.0 from path
Installing myiis         >= 0.0.0 from path
Installing mychef_client >= 0.0.0 from path
Installing chef-client   11.5.0
Installing cron          6.3.0
Installing logrotate     2.2.2

Lockfile written to /home/centos/chef-repo/company_web.lock.json
Policy revision id: 2daf2bd72e485c148340b712bcd139783f17f3dc5d34877383dc367812417a04
```

GL: Push the Policyfile



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

```
Uploading policy company_web (2daf2bd72e) to policy group prod
Using    apache        0.2.0  (6ef4b917)
Using    company_web   0.1.0  (4353e774)
Using    myiis         0.2.1  (cd0db3ed)
Uploaded chef-client   11.5.0 (7cb128f1)
Uploaded cron          6.3.0  (b2c49b29)
Uploaded logrotate     2.2.2  (bd20a5c5)
Uploaded mychef_client 0.1.0  (c9d53cd1)
```

GL: Converge the Linux Web Node

```
$ knife ssh 'name:node1' -x centos -i ~/aws.pem 'sudo chef-client'
```

```
…
ec2-34-196-104-17.compute-1.amazonaws.com Using policy 'company_web' at
revision '7f50faaaf9eceb35f72358c38a71b86c7fb7df8f49fee245827eb4613d919827’
…
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   - company_web (0.1.0)
ec2-34-196-104-17.compute-1.amazonaws.com   - cron (6.2.1)
ec2-34-196-104-17.compute-1.amazonaws.com   - myiis (0.2.1)
ec2-34-196-104-17.compute-1.amazonaws.com   - logrotate (2.2.0)
ec2-34-196-104-17.compute-1.amazonaws.com   - mychef_client (0.1.0)
ec2-34-196-104-17.compute-1.amazonaws.com   - chef-client (11.2.0)
…
```

GL: Verify chef-client is Running (Linux)

```
$ knife ssh 'name:node1' -x centos -i ~/aws.pem "ps
awux | grep chef-client"
```

```
ec2-34-196-104-17.compute-1.amazonaws.com root      4710  0.4  5.9 298480 60604 ?
Ss   16:12   0:01 /opt/chef/embedded/bin/ruby --disable-gems /usr/bin/chef-client -c
/etc/chef/client.rb -i 1800 -s 300

ec2-34-196-104-17.compute-1.amazonaws.com chef      4721  0.0  0.1 113180  1608
pts/0    Ss+  16:18   0:00 bash -c ps awux | grep chef-client
ec2-34-196-104-17.compute-1.amazonaws.com chef      4737  0.0  0.0 112708   972
pts/0    S+   16:18   0:00 grep chef-client
```

Notice that the interval is set to 1800 seconds.
- number of seconds between chef-client daemon runs.

https://docs.chef.io/ctl_chef_client/

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
```

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

GL: Update the Policyfile

```
$ 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   11.2.0
Using      cron          6.2.1
Using      logrotate     2.2.0
Lockfile written to /Users/sdelfante/chef-repo/company_web.lock.json
Policy revision id
```

GL: Push the Policyfile to Prod

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

```
Uploading policy company_web (891f323cb2) to policy group prod
Using    apache        0.1.0  (1388ab3a)
Using    chef-client   11.2.0 (0b49a3a8)
Using    company_web   0.1.0  (c1b26cb5)
Using    cron          6.2.1  (08676b5c)
Using    logrotate     2.2.0  (53e09234)
Using    myiis         0.2.1  (cd0db3ed)
Uploaded mychef_client 0.1.0  (10d082a4)
```

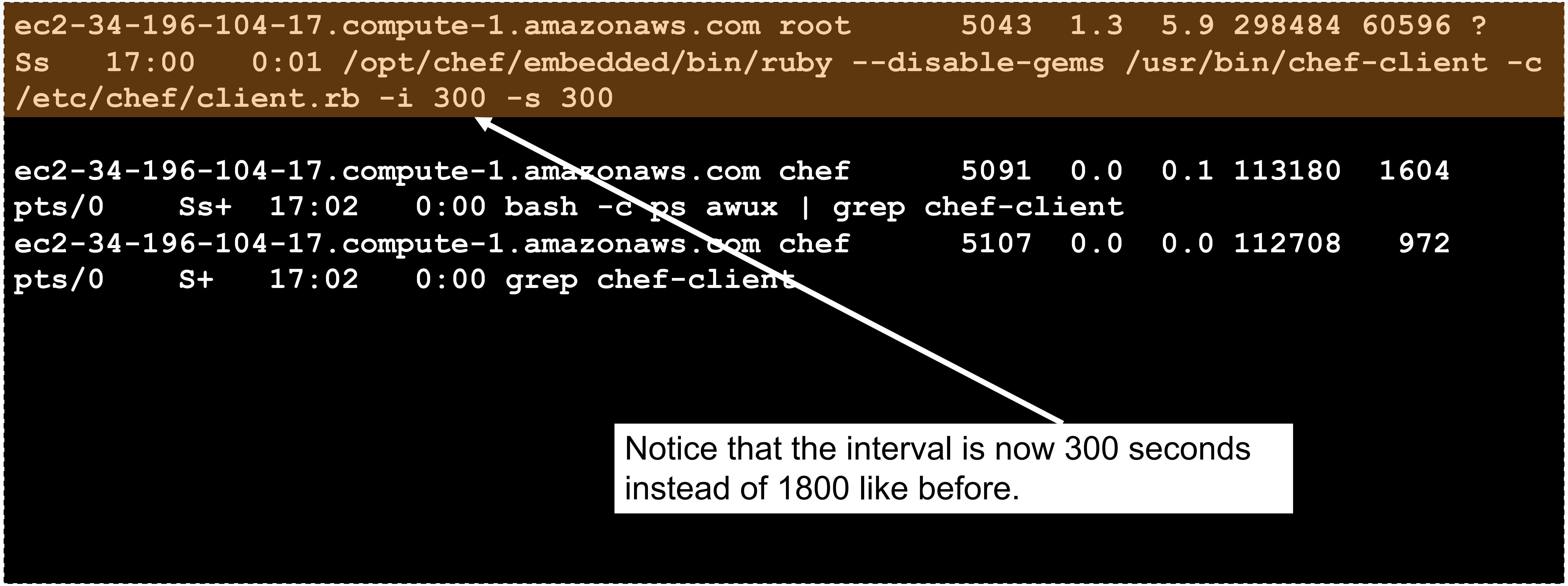
GL: Converge the Linux Web Node

```
$ knife ssh 'name:node1' -x centos -i ~/aws.pem '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 (11.2.0)
ec2-34-196-104-17.compute-1.amazonaws.com   - cron (6.2.1)
ec2-34-196-104-17.compute-1.amazonaws.com   - myiis (0.2.1)
ec2-34-196-104-17.compute-1.amazonaws.com   - logrotate (2.2.0)
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  2019-07-31
16:12:34.501755564 +0000
```

GL: Verify chef-client is Running (Linux)

```
$ knife ssh 'name:node1' -x centos -i ~/aws.pem "ps
awux | grep chef-client"
```



```
ec2-34-196-104-17.compute-1.amazonaws.com root      5043  1.3  5.9 298484 60596 ?
Ss   17:00   0:01 /opt/chef/embedded/bin/ruby --disable-gems /usr/bin/chef-client -c
/etc/chef/client.rb -i 300 -s 300

ec2-34-196-104-17.compute-1.amazonaws.com chef      5091  0.0  0.1 113180  1604
pts/0    Ss+  17:02   0:00 bash -c ps awux | grep chef-client
ec2-34-196-104-17.compute-1.amazonaws.com chef      5107  0.0  0.0 112708   972
pts/0    S+   17:02   0:00 grep chef-client
```

Notice that the interval is now 300 seconds instead of 1800 like before.

Q&A

What questions can we help you answer?

- Chef Supermarket
- Wrapper Cookbooks
- chef-client Cookbook

CHEF™