What is KitchenCI?

- Answer : Kitchen comes at the process of software development with an approach that embraces the idea that writing the tests first , watching them fail , and then writing the code to make them pass is a great way to go.

Installing for Kitchen

- chef —version
- VBoxManage —version
- vagrant —version

Getting Help

- kitchen help
- kitchen help diagnose

Creating a Cookbook

- chef generate cookbook git_cookbook
- cd git_cookbook

The .kitchen.yml

- To see the results of our work, let's run the kitchen list subcommand

Instances

What is an Instance ?

Answer : A kitchen instance is a combination of a Suite and a Platform as laid out in your .kitchen.yml file. Kitchen has auto-named our only instance by combining the Suite name("default") and the Platform name("Ubuntu-16.04") into a form that is safe for DNS and hostname records, namely "default-ubuntu-16.04".

Command:

—> kitchen create default-ubuntu-1604

```
[sh-3.2# kitchen list
Instance              Driver   Provisioner   Verifier  Transport  Last Action     Last Error
default-ubuntu-1604   Vagrant  ChefZero      Inspec    Ssh        Created         <None>
default-centos-73     Vagrant  ChefZero      Inspec    Ssh        <Not Created>   <None>
```

Writing a Recipe?

Answer : cd recipes/default.rb

- package "git"

Kitchen converge?

Answer : A converge will leave the machine running and kitchen automatically uploads changes each converge so that one can iterate rapidly on configuration code. A lot of time and effort has gone into ensuring that the exit code of kitchen is always appropriate.

—> kitchen converge default-ubuntu-1604

Manually Verifying?

```
[sh-3.2# which git
/usr/bin/git
[sh-3.2# git --version
git version 2.11.0 (Apple Git-81)
```

Answer :

Writing a Test ?

Answer : test/smoke/default/default_test.rb

```
[sh-3.2# cat default_test.rb
# # encoding: utf-8

# Inspec test for recipe git_cookbook::default

# The Inspec reference, with examples and extensive documentation, can be
# found at http://inspec.io/docs/reference/resources/

#unless os.windows?
  # This is an example test, replace with your own test.
#  describe user('root'), :skip do
#    it { should exist }
#  end
#end

# This is an example test, replace it with your own test.

#describe port(80), :skip do
#  it { should_not be_listening }
#end

describe package('git') do
  it { should be_installed }
end
```

-
- **Kitchen verify**

```
sh-3.2# kitchen verify default-ubuntu-1604
-----> Starting Kitchen (v1.16.0)
-----> Verifying <default-ubuntu-1604>...
       Loaded tests from test/smoke/default

Profile: tests from test/smoke/default
Version: (not specified)
Target:  ssh://vagrant@127.0.0.1:2201


    System Package
       ✔ git should be installed

Test Summary: 1 successful, 0 failures, 0 skipped
       Finished verifying <default-ubuntu-1604> (0m0.20s).
```
Answer: -----> Kitchen is finished. (0m1.17s)

```
sh-3.2# kitchen list
Instance              Driver   Provisioner   Verifier  Transport  Last Action     Last Error
default-ubuntu-1604   Vagrant  ChefZero      Inspec    Ssh        Verified        <None>
default-centos-73     Vagrant  ChefZero      Inspec    Ssh        <Not Created>   <None>
```

- **Kitchen test**
  - **kitchen test default-ubuntu-1604 —> It will destroy everything so please convergence instead of using test.**

```
Profile: tests from test/smoke/default
Version: (not specified)
Target:  ssh://vagrant@127.0.0.1:2201


    System Package
       ✔ git should be installed

Test Summary: 1 successful, 0 failures, 0 skipped
       Finished verifying <default-ubuntu-1604> (0m0.22s).
-----> Destroying <default-ubuntu-1604>...
       ==> default: Forcing shutdown of VM...
       ==> default: Destroying VM and associated drives...
       Vagrant instance <default-ubuntu-1604> destroyed.
       Finished destroying <default-ubuntu-1604> (0m4.60s).
       Finished testing <default-ubuntu-1604> (0m57.66s).
  -----> Kitchen is finished. (0m58.61s)
```

- kitchen list - You will see not created

Adding a Platform

- Now come and work on the Centos
- Let's see how CentOS runs our cookbook
  - kitchen verify 73
  - kitchen list
  - kitchen destroy —> Any kitchen subcommand without an instance argument will apply to all instances—> This will destroy CentOS 7.3
-