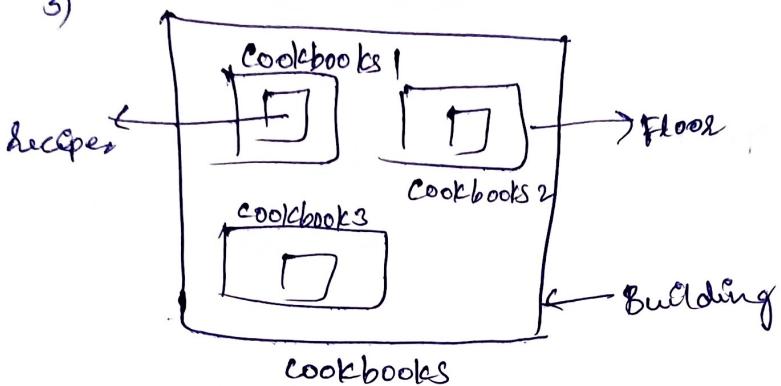


1) Create Linux and install Chef-Workstation

2) Create directory `mkadir cookbooks`

3)



4) To understand logic

Suppose 1 Building we have floors like

1st floor (1st cookbook), 2nd floor (2nd cookbook) . . .

5) Process to install Chef-Workstation

Yum install -y <Chef-station Linux>

Which chef

chef --version

6) `mkadir cookbook`

chef generate cookbook <cookbook-name>

7) Yum install tree -y

8) tree ⇒ run a command to see clear view of
Folders and files

9) Suppose & create cookbook

chef generate cookbook <test cookbook>

10) `ed testcookbook`

chef generate recipe <test recipe>

11)

`cd ..`

`vi testcookbook/recipes/test recipe.rb`

12)

(

)

(

)

(

(

(

(

(

(

(

(

(check) To test if error or not

(13) chef spec ruby -c testcookbook [or epos] testrecipewh

(run) (14) chef-client -z "recipe[testcookbook::testrecipe]"

resource

- 1) package
- 2) file
- 3) user
- 4) template
- 5) cron
- 6) directory
- 7) execute
- 8) cookbook-file
- 9) service
- 10) user

- 1) directory
- 2) file
- 3) package
- 4) service
- 5) user
- 6) group
- 7) cookbook-file
- 8) template
- 9) execute
- 10) cron

(15) Login Linux machine

ec2-user

- (16) sudo su -
- (17) whoami
- (18) whoami \$address
- (19) whoami memory/total
- (20) whoami cpu/0/wh2
- (21) cd cookbook2
- (22) chef generate cookbook <cookbookname>
- (23) cd cookbookname
- (24) chef generate recipe <recipename>
- (25) cd ..
- (26) vi <cookbookname>/recipes/<recipename>.rb
- (27) chef spec ruby -c <cookbookname>/recipes/<recipename>.rb
- (28) chef-client -z "recipe[cookbookname::recipename]"

~~# Insert Linux command, create user and group~~

on
in

②9 Insert Linux commands, Create user & group

③0 execute "run a script" do
command <<-EOH
mkdir /bashra
touch /bashrafile
EOH
end

EOH ⇒ End of here
End of block

④0

③1 user "Bashra" do
action: create
end

③2 group "developer" do
action: create
member 'bashra'
append true
end

③3 # To Run Multiple recipe in two ways

2)

3)

③4 chef-client -z "recipe [cookbook1]name:: recipename],
recipe [cookbook2]name:: recipename"]"

4)

5)

③5 cd cookbook

6)

③6 chef generate cookbook <cookbook name> and cd cookbook name

③7 chef generate recipe <recipename>

③8 After creating of recipes Go to recipes folder
You may notice one default.rb

③9 vs default.rb

Login Amazon Linux machine

ec2-user

sudo su -

[ec2-user] # ls

[ec2-user] # cd cookbooks/

[cookbooks] # ls

O/p → apache-cookbook test-cookbook

Now, Open google chrome → search chef.io (manage.chef)

→ Create One account

Go to chef account → click on Organisation

→ Starter kit → Download starter kit

→ Open the download content → unzip → chef-repo

Now download "WinSCP" → login with ec2 credentials

Now drag and drop "chef-repo" folder from

Window to Lintax

Now, Open workstation in AWS again

[cookbooks] # ls

[cookbooks] # cd ..

[ec2-user] # ls

O/p → chef-repo cookbooks roles

[ec2-user] # cd chef-repo

[chef-repo] # ls -a

O/p → .chef cookbook roles

```
[chef-repo]# cd .chef/
```

```
[.chef]# ls
```

O/p → knife.rb bashar.pem

```
[.chef]# cat knife.rb
```

You will get URL of chef server

```
[.chef]# cd ..
```

```
[chef-repo]# knife ssl check
```

O/p → successfully verified certificate from api

① Attach node to chef-server

② Create one linux machine (Node), launch in same Availability Zone

③ Advance details

#!/bin/bash

Sudo su-

Yum update -y

Now go to chef-workstation

```
[chef-repo]#
```

```
knife bootstrap 172.31.4.89 --ssh-user ec2-user  
--sudo -i node-key.pem -N Node1
```

```
[chef-repo]# knife node list
```

upload cookbook from workstation to chef server

[chef-repo]# knife cookbook upload <cookbookname>

[chef-repo]# knife cookbook list

[chef-repo]# knife node run_list set node1

"recipe[<cookbookname>::<reipename>]"

Suppose I have a lot of nodes okay !!

Note:- I have a lot of cookbook

eg:- windows server

linux server

Web server

I have set only one cookbook bcoz I don't want windows server cookbook and linux server cookbook and soon...

Important

① knife cookbook upload <cookbookname>

② knife node run_list set node1

"recipe[<cookbookname>::<reipename>]"

③ chef-client

(chef-repo) # knife node show node1

0/8 → Run list :- recipe[apache-cookbook :: apache-recipe]

To automatic in Node

vi /etc/crontab

* * * * * root chef-client

!@w!

Or else while creating a node

Advance details

```
#!/bin/bash
sudo su -
yum update -y
echo "* * * * * root chef-client" >> /etc/crontab
```

Create Role in chef

knife cookbook list

knife cookbook delete <cookbookname> -y

knife node delete <nodename> -y

knife node list

knife chef-list

knife client-list

knife client delete <client name> -y

knife role list

knife role delete <role name> -y

Create roles

[chef-repo] # ls

o/p .chef roles>

[chef-repo] # cd roles/

[roles] # ls

o/p starter.rb

[roles] # vi devops.rb

name "devops"

description "role created"

run_list "recipe[cookbookname]: ~<recipename>"

: war!

① knife roles from file roles/devops.rb

② knife node run_list set/add node1 "roles[devops]"

Chef cheat sheet

local machine

- ① chef generate cookbook <cookbookname>
- ② chef generate recipe <recipename>
- ③ chef spec ruby -c cookbook/recipes/recipe.rb
(os)
- ④ chef exec ruby -c cookbook/recipes/recipe.rb
- ⑤ chef-client -z <recipename> [<cookbookname>::<recipename>]

Important

cd chef-repo \Rightarrow cd cookbooks

- ① chef generate cookbook <cookbookname>

- ② cd <cookbookname>

- ③ chef generate recipe <recipename>

- ④ cd ..

- ⑤ chef exec ruby -c cookbook/recipes/recipe.rb

- ⑥ knife cookbook upload <cookbookname>

- ⑦ knife node run-list add node { "recipe[<cookbookname>::<recipename>]" }

- ⑧ roles

- ⑨ cd roles

- ⑩ create role

- ⑪ v devops.rb

- ⑫ name "rolename (devops)"

description "This is a role"

run_list "recipe [<cookbookname>::<recipename>]"

- ⑬ knife role from file roles/devops.rb

- ⑭ knife node run-list add node { "roles[<rolename>]" }

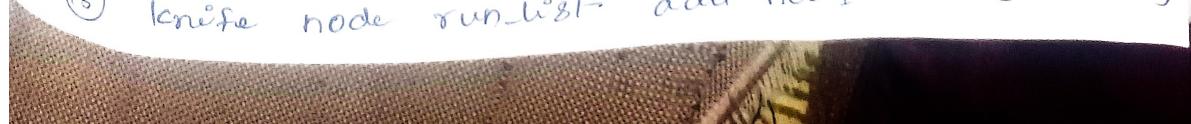
Chef cheat sheet

local machine

- ① chef generate cookbook <cookbookname>
- ② chef generate recipe <recipename>
- ③ chef spec ruby -c cookbook/recipes/recipe.rb
(optional)
- ④ chef exec ruby -c cookbook/recipes/recipe.rb
- ⑤ chef-client -z <recipe> [<cookbookname>; <recipename>]

Important
cd chef-repo \Rightarrow cd cookbooks

- ① chef generate cookbook <cookbookname>
- ② cd <cookbookname>
- ③ chef generate recipe <recipename>
- ④ cd ..
- ⑤ chef exec ruby -c cookbook/recipes/recipe.rb
- ⑥ knife cookbook upload <cookbookname>
- ⑦ knife node run_list add node "recipe[<cookbookname>; <recipename>]"
- ⑧ roles
- ⑨ cd roles
- ⑩ create role
- ⑪ vi devops.rb
- ⑫ name "rolename (devops)"
description "This is a role"
- ⑬ run_list "recipe [<cookbookname>; <recipename>]"
- ⑭ knife role from file roles/devops.rb
- ⑮ knife node run_list add node "roles[<rolename>]"



- (16) chef-client [apply on nodes] this is manual
- (17) vi /etc/crontab [To automate go to this path]
* * * * * root chef-client
- (18) knife node list
- (19) knife node delete <node name> -y
- (20) knife client list
- (21) knife client delete <client name> -y
- (22) knife cookbook list
- (23) knife cookbook delete <cookname> -y
- (24) knife role list
- (25) knife role delete <role name> -y
- (26) knife node show node1 [run list in node]
- (27) knife bootstrap <ip address> --ssh-user ec2-user
-N <Node_Name> --sudo -i <key.pem>