DevOps
Chef for Configuration Management(2)
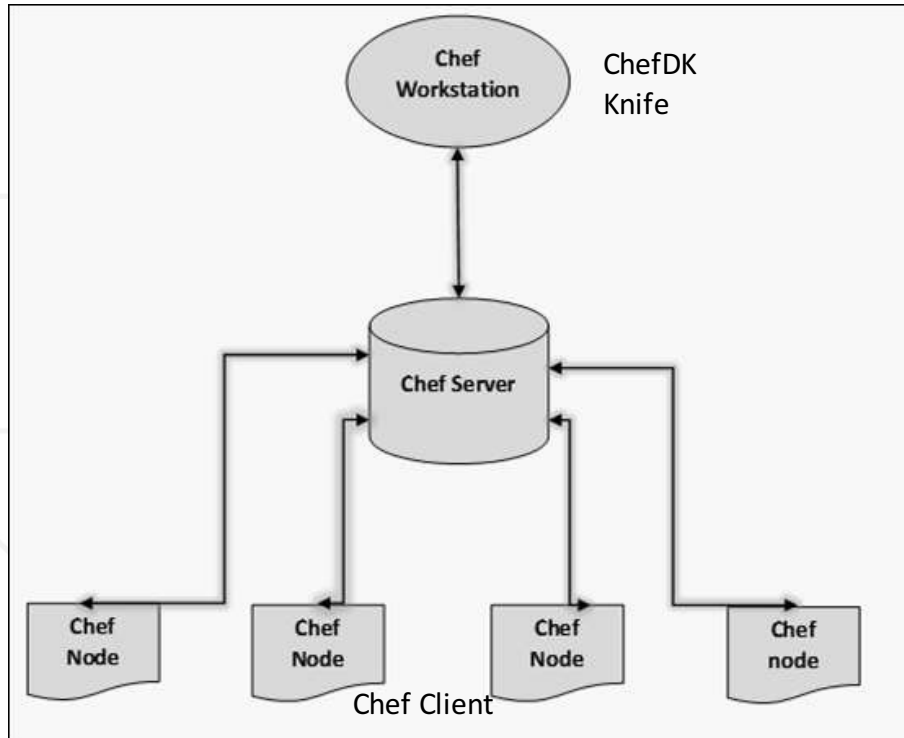
Chef is a powerful automation platform that transforms infrastructure into code.

Pull
Chef Recipe

ChefDK

workstation

Chef server
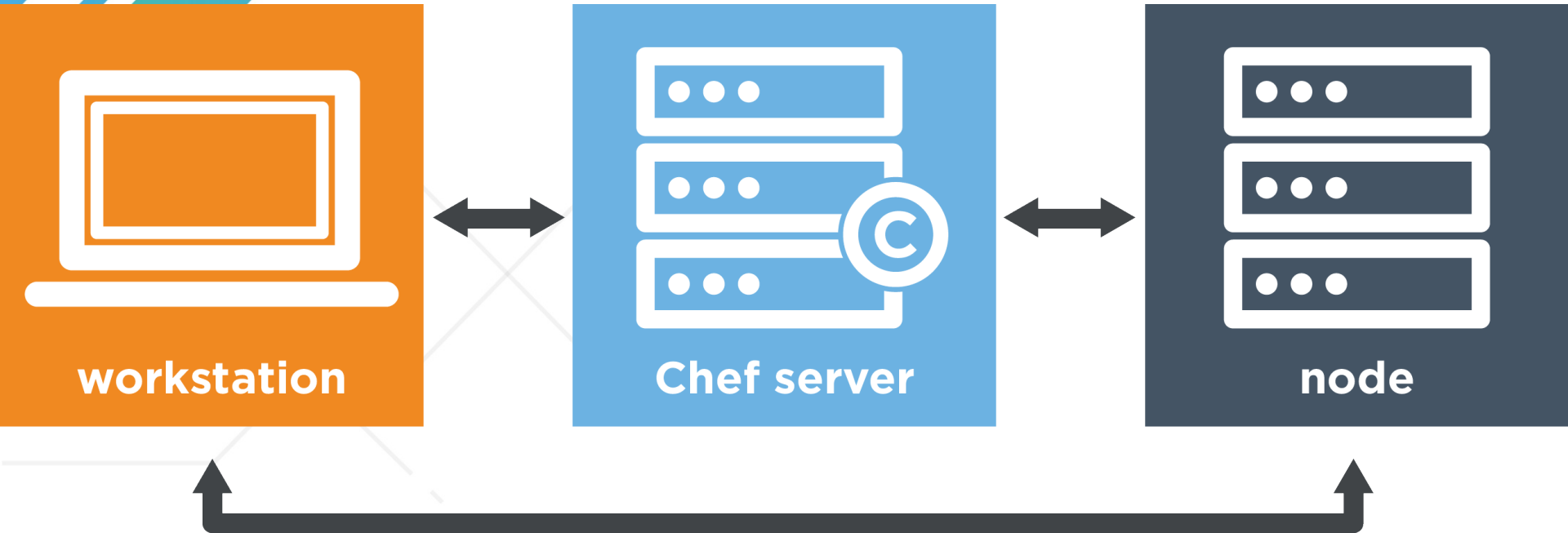
node

Running of chef-client command by nodes is called **convergence**

Automated chef-client run:
sudo chef-client -i 1800 &

Login to the node and run chef-client

```
root@chefnode1:~# chef-client
Starting Chef Client, version 11.8.2
resolving cookbooks for run list: ["sree"]
Synchronizing Cookbooks:
  - sree
Compiling Cookbooks...
Converging 1 resources
Recipe: sree::default
  * file[/root/chef-test1.txt] action create
    - create new file /root/chef-test1.txt
```

Chef: Test Node Setup: Check node details using knife

```
root@chefworkstation:~/chef-repo# knife node list
chefnode1
chefnode2
root@chefworkstation:~/chef-repo# knife node list -w
chefnode1: https://api.chef.io/organizations/schogini1/nodes/chefnode1
chefnode2: https://api.chef.io/organizations/schogini1/nodes/chefnode2
root@chefworkstation:~/chef-repo#
```

workstation

Chef: Test Node Setup: Check node details using knife

```
root@chefworkstation:~/chef-repo# knife node show chefnode1
Node Name:     chefnode1
Environment:   _default
FQDN:          chefnode1
IP:            10.0.2.15
Run List:      recipe[sree]
Roles:
Recipes:       sree
Platform:      ubuntu 14.04
Tags:
```

workstation

Chef: Test Node Setup: Check node details using knife

List run list of a node

```
root@chefworkstation:~/chef-repo# knife node show chefnode1 -r
chefnode1:
  run_list: recipe[sree]
```

Long listing of detailed node info
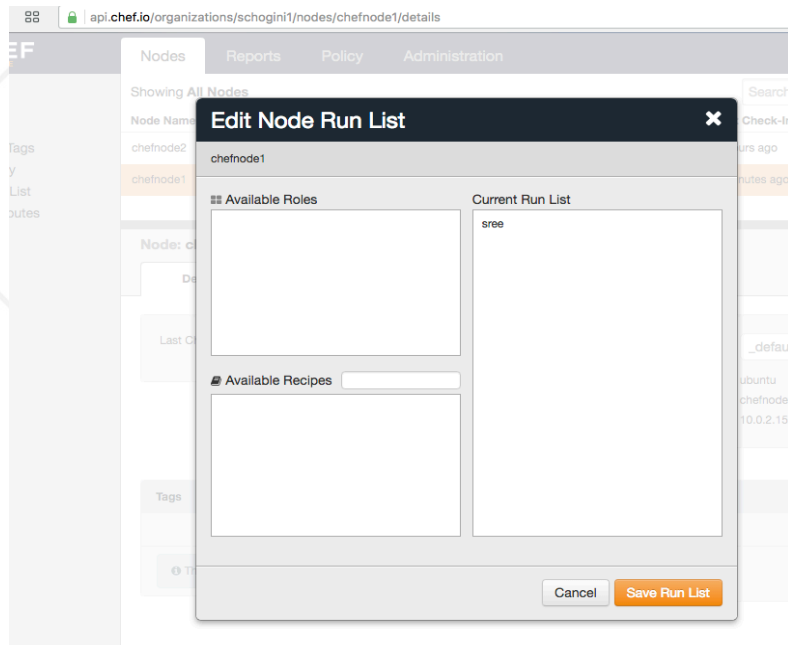
```
root@chefworkstation:~/chef-repo# knife node show chefnode1 -l
```

workstation

Login to the workstation and run the knife command to add run_list

root@chefworkstation:~/chef-repo# **knife node run_list add chefnode1 "recipe[sree]"**

```
root@chefworkstation:~/chef-repo# knife search node "*"
2 items found

Node Name:      chefnode2
Environment:    _default
FQDN:           chefnode2
IP:             10.0.2.15
Run List:
Roles:
Recipes:
Platform:       ubuntu 14.04
Tags:

Node Name:      chefnode1
Environment:    _default
FQDN:           chefnode1
IP:             10.0.2.15
Run List:       recipe[sree]
Roles:
Recipes:        sree
Platform:       ubuntu 14.04
Tags:
```

workstation

```
root@chefworkstation:~/chef-repo# knife search node "ubuntu:*"
2 items found

Node Name:      chefnode2
Environment:    _default
FQDN:           chefnode2
IP:             10.0.2.15
Run List:
Roles:
Recipes:
Platform:       ubuntu 14.04
Tags:

Node Name:      chefnode1
Environment:    _default
FQDN:           chefnode1
IP:             10.0.2.15
Run List:       recipe[sree]
Roles:
Recipes:        sree
Platform:       ubuntu 14.04
Tags:
```

workstation

18008338228   +65 31586636   +1(973) 598-3969   44 203-808-4216   www.cognixia.com • bdg@cognixia.com

```
root@chefworkstation:~/chef-repo# knife search node 'recipes:sree'
1 items found

Node Name:      chefnode1
Environment:    _default
FQDN:           chefnode1
IP:             10.0.2.15
Run List:       recipe[sree]
Roles:
Recipes:        sree
Platform:       ubuntu 14.04
Tags:
```

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

workstation

Chef: A Recipe to Create a Web Server

```
GNU nano 2.2.6                                    File: recipes/default2.rb

#
# Cookbook:: apache
# Recipe:: default
#
# Copyright:: 2017, The Authors, All Rights Reserved.

time =   Time.new.strftime("%Y-%m-%d %H:%M:%S")
node = Chef::Config[:node_name]

package "apache2" do
        action  :install
end

template "/var/www/html/index.html" do
        source "index.html.erb"
        action  :create
        mode "0644"
        variables(
                :title  => "Sree",
                :desc   => " DevOps Pro!",
                :time => time,
                :node => node
        )
end

service "apache2" do
        action  [ :enable, :start ]
end
```

Chef: A Recipe to Create a Web Server

```
GNU nano 2.2.6                    File: templates/default/index.html.erb

<!DOCTYPE html>
<html>
<head>
        <title><%= @title %></title>
</head>
<body>
<h1><%= @title %> from <%= node %></h1>
<h3>The <%= @desc %></h3>
<hr>
<h3>I was created @ <%= @time %> by <%= node %></h3>
</body>
</html>
```

Chef: Environments: What is it?

An environment is a way to map an organization's real-life workflow to what can be configured and managed when using Chef server.

Chef **Organization** starts with a **_default** environment, which cannot be modified (or Deleted)
Additional **custom environments** can be created.
Eg: production, staging, testing, development etc.

An environment is associated with one (or more) **cookbook versions** (in general)

Nodes, roles, run-lists, cookbooks (and cookbook versions), and attributes specific to an organization can only be associated with a custom environment!

Attributes in env will override cookbook attribute

Environments

Development Nodes

Staging Nodes

Production Nodes

Cookbook 1.5.1

Cookbook 1.3.1

Cookbook 1.2.1

An environment is associated with one (or more) **cookbook versions** (in general)

```
cd cookbooks/sree
cat metadata.rb

name 'sree'
maintainer 'Sreeprakash Neelakantan'
maintainer_email 'sree@xyx.com'
license 'All Rights Reserved'
description 'Installs/Configures sree'
long_description 'Installs/Configures cookbook sree'
version "0.1.1"
```

workstation

A file for each custom environment must exist in the environments/ subdirectory of the chef-repo. (It should be created.)

Sample environment syntax: environments/dev.rb

```
GNU nano 2.2.6                    File: environments/dev.rb

name "dev"
description "Development environment"
cookbook "sree", "= 0.1.1"
```

workstation

Chef: Environments: How to upload an Environment

To upload an environment to Chef Server from /root/chef-repo
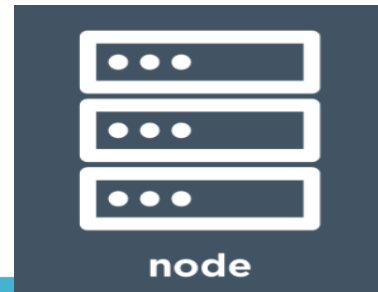knife environment from file environments/dev.rb
To see the list of environments
knife environment list -w

```
_default:    https://api.chef.io/organizations/schogini1/environments/_default
[root@chefworkstation:~/chef-repo# knife environment from file environments/dev.rb
Updated Environment dev
[root@chefworkstation:~/chef-repo# knife environment list -w
_default:    https://api.chef.io/organizations/schogini1/environments/_default
dev:         https://api.chef.io/organizations/schogini1/environments/dev
```

workstation

1. ** Add a line like this to the */etc/chef/client.rb* file on the node. Chef-client will then, pick it up: *environment "dev"*

2. At the time of bootstrapping you can use the –E option to set the environment. Example: *knife bootstrap chefnode1 -x root -P sree1234 -N chefnode1 **–E "dev"***

3. You can manually set it using the knife command:
   knife node environment_set chefnode1 dev

Chef: Environments: Some commands

knife environment list -w
knife environment compare dev
knife environment compare dev prod
knife environment compare --all
knife environment delete dev
knife environment show dev

workstation

nano roles/webserver.rb

```
GNU nano 2.2.6                          File: roles/webserver.rb

name "webserver"
  description "A webserver role"
  run_list "recipe[apache]", "recipe[sree]"
```

workstation

Chef: Roles: Upload a role to Chef Server/Organization

```
[root@chefworkstation:~/chef-repo# nano roles/webserver.rb
[root@chefworkstation:~/chef-repo# knife role from file roles/webserver.rb
Updated Role webserver
[root@chefworkstation:~/chef-repo# knife role list -w
webserver: https://api.chef.io/organizations/schogini1/roles/webserver
```

workstation

```
[root@chefworkstation:~/chef-repo# knife node run_list add chefnode1 "role[webserver]"
chefnode1:
   run_list:
     recipe[sree]
     role[webserver]
root@chefworkstation:~/chef-repo#
```

Chef: Extra

```
[root@chefworkstation:~/chef-repo# knife node show chefnode1 -r
chefnode1:
   run_list:
      recipe[sree]
      role[webserver]
      recipe[apache]
[root@chefworkstation:~/chef-repo# knife node show chefnode2 -r
chefnode2:
   run_list:
```

Chef: Extra

```
root@chefworkstation:~/chef-repo# knife role show webserver
chef_type:              role
default_attributes:
description:            A webserver role
env_run_lists:
json_class:             Chef::Role
name:                   webserver
override_attributes:
run_list:
   recipe[apache]
   recipe[sree]
```

Chef: Extra

```
root@chefworkstation:~/chef-repo/cookbooks# chef generate cookbook apache_attrib
Generating cookbook apache_attrib
- Ensuring correct cookbook file content
- Committing cookbook files to git
- Ensuring delivery configuration
- Ensuring correct delivery build cookbook content
- Adding delivery configuration to feature branch
- Adding build cookbook to feature branch
- Merging delivery content feature branch to master

Your cookbook is ready. Type `cd apache_attrib` to enter it.

There are several commands you can run to get started locally developing and testing your cookbook.
Type `delivery local --help` to see a full list.

Why not start by writing a test? Tests for the default recipe are stored at:

test/smoke/default/default_test.rb

If you'd prefer to dive right in, the default recipe can be found at:

recipes/default.rb


root@chefworkstation:~/chef-repo/cookbooks# ls -l
total 20
drwxr-xr-x 9 root root 4096 Aug 17 05:55 apache
drwxr-xr-x 7 root root 4096 Aug 18 06:48 apache_attrib
-rw-r--r-- 1 root root  588 Aug 10 08:30 chefignore
drwxr-xr-x 7 root root 4096 Aug 10 09:21 sree
drwxr-xr-x 6 root root 4096 Aug 10 08:30 starter
```

Chef: Extra

```
[root@chefworkstation:~/chef-repo/cookbooks/apache_attrib/recipes# mkdir -p ../templates/default
[root@chefworkstation:~/chef-repo/cookbooks/apache_attrib/recipes# mv index.html.erb ../templates/default/
[root@chefworkstation:~/chef-repo/cookbooks/apache_attrib/recipes# cd ..
[root@chefworkstation:~/chef-repo/cookbooks/apache_attrib# cd ..
[root@chefworkstation:~/chef-repo/cookbooks# cd ..
[root@chefworkstation:~/chef-repo# knife cookbook upload "apache_attrib"
Uploading apache_attrib  [0.1.0]
Uploaded 1 cookbook.
```

Chef: Extra

```
[root@chefworkstation:~/chef-repo/roles# nano testserver.rb
```

```
GNU nano 2.2.6                                    File: testserver.rb


name "testserver"
  description "A test webserver role"
  run_list "recipe[apache_attrib]", "recipe[sree]"
```

```
[root@chefworkstation:~/chef-repo# knife role from file roles/testserver.rb
Updated Role testserver
[root@chefworkstation:~/chef-repo# knife role list
testserver
webserver
```

Chef: Extra

```
[root@chefworkstation:~/chef-repo# knife node run_list set chefnode2 "role[testserver]"
chefnode2:
   run_list: role[testserver]
```

```
[root@chefworkstation:~/chef-repo# knife node show chefnode1 -r
chefnode1:
   run_list:
      recipe[sree]
      role[webserver]
      recipe[apache]
[root@chefworkstation:~/chef-repo# knife node show chefnode2 -r
chefnode2:
   run_list: role[testserver]
```

```
[root@chefworkstation:~/chef-repo# knife node run_list remove chefnode2 "role[testserver]"
chefnode2:
   run_list:
```

Chef: Extra

```
[root@chefworkstation:~/chef-repo# knife node show chefnode2
Node Name:     chefnode2
Environment: _default
FQDN:          chefnode2
IP:            10.0.2.15
Run List:
Roles:         testserver
Recipes:       apache_attrib, sree
Platform:      ubuntu 14.04
Tags:
[root@chefworkstation:~/chef-repo# knife node run_list remove chefnode2 "role[testserver]"
WARNING: role[testserver] is not in the run list
chefnode2:
  run_list:

[root@chefworkstation:~/chef-repo# knife node show chefnode2
Node Name:     chefnode2
Environment: _default
FQDN:          chefnode2
IP:            10.0.2.15
Run List:
Roles:
Recipes:
Platform:      ubuntu 14.04
Tags:
[root@chefworkstation:~/chef-repo# knife node run_list set chefnode2 "role[webserver]"
chefnode2:
  run_list:  role[webserver]
```
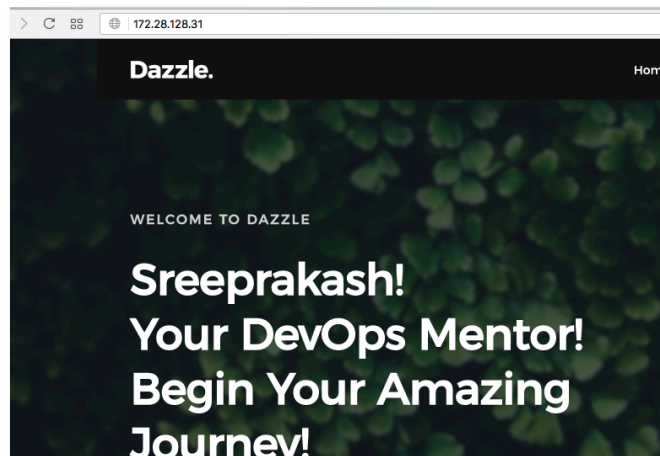
Chef: Extra

```
[root@chefworkstation:~/chef-repo# knife node run_list set chefnode2 "role[testserver]"
chefnode2:
    run_list: role[testserver]
[root@chefworkstation:~/chef-repo# knife node run_list set chefnode2 "role[webserver]"
chefnode2:
    run_list: role[webserver]
```

172.28.128.31

Sree from node[chefnode2]

The DevOps Pro!

I was created @ 2017-08-18 08:29:53 by node[chefnode2]

172.28.128.31

Dazzle.   Home

WELCOME TO DAZZLE

**Sreeprakash!
Your DevOps Mentor!
Begin Your Amazing
Journey!**

For Additional Information Read This
https://www.linode.com/docs/applications/configuration-management/creating-your-first-chef-cookbook

Build a Chef Recipe to Auto Provision a Complete Website.

You should copy the index.html as index.html.erb in the templates area and add the attributes. i.e. The recipe should use attributes to replace template variables.

Note: Regarding the theme that I used, its just a theme I downloaded from the internet. You can download any free theme from the internet and create a template from the index file of that theme, copy the remaining files and use it.