Lab: Define a Load Balancer Role

- ❑ Create a role named 'load-balancer' that has the run list 'recipe[myhaproxy]'
- ❑ Upload the role with 'knife role from file'
- ❑ Set the load-balancer node's run list to be "role[load-balancer]"
- ❑ Converge your load balancer node

Lab: Create the load-balaner.rb File

~/chef-repo/roles/load-balancer.rb

```
name 'load-balancer'
description 'Load Balancer Role'
run_list 'recipe[myhaproxy]'
```

Lab: Upload the web.rb File

```
$ knife role from file roles/load-balancer.rb
```

```
Updated Role load-balancer!
```

Lab: Verify the Role on the Chef Server

```
$ knife role list
```

```
load-balancer
web
```


CHEF

Lab: Verify Specific Information About the Role

```
$ knife role show load-balancer
```

```
chef_type:            role
default_attributes:
description:          Load Balancer Role
env_run_lists:
json_class:           Chef::Role
name:                 load-balancer
override_attributes:
run_list:             recipe[myhaproxy]
```

Lab: Set node1's Run List

```
$ knife node run_list set load-balancer "role[load-balancer]"
```

```
load-balancer:
  run_list: role[load-balancer]
```


CHEF

Verify the Run List

```
$ knife node show load-balancer
```

```
Node Name:   load-balancer
Environment: _default
FQDN:        load-balancer
IP:          10.0.2.15
Run List:    role[load_balancer]
Roles:
Recipes:     myhaproxy, myhaproxy::default, haproxy::default, haproxy::install_package
Platform:    centos 7.2
Tags:
```


CHEF

Login to Load Balancer

```
$ vagrant ssh load-balancer
```

```
Last login: Sat Dec 31 02:59:27 2016 from 10.0.2.2
[vagrant@load-balancer ~]$
```

Converge the Load Balancer

```
[vagrant@load-balancer ~]$ sudo chef-client
```

```
Starting Chef Client, version 12.17.44
resolving cookbooks for run list: ["myhaproxy"]
Synchronizing Cookbooks:
  - myhaproxy (0.1.0)
  - haproxy (2.0.0)
  - build-essential (7.0.3)
  - seven_zip (2.0.2)
  - windows (2.1.1)
  - ohai (4.2.3)
....
```

Return to your Workstation

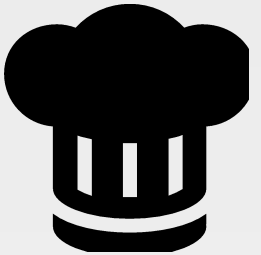
```
[vagrant@load-balancer ~]$ exit
```

```
logout
Connection to 127.0.0.1 closed.
```

Verify the Run List

```
$ knife node show load-balancer
```

```
Node Name:   load-balancer
Environment: _default
FQDN:        load-balancer
IP:          10.0.2.15
Run List:    role[load-balancer]
Roles:       load-balancer
Recipes:     myhaproxy, myhaproxy::default, haproxy::default, haproxy::install_package
Platform:    centos 7.2
Tags:
```



Roles for Everyone

We will give our nodes a role to better describe them and so we can configure them in a similar manner.

Objective:

- ✓ Give our load balancer node a "load_balancer" Role
- ✓ Give our web nodes a "web" Role

CHEF

Discussion

What are the benefits of using roles? What are the drawbacks?

Roles can contain roles. How many of these nested roles would make sense?