Environments

Using Environments to Reflect Organization Patterns and Workflow

CHEF

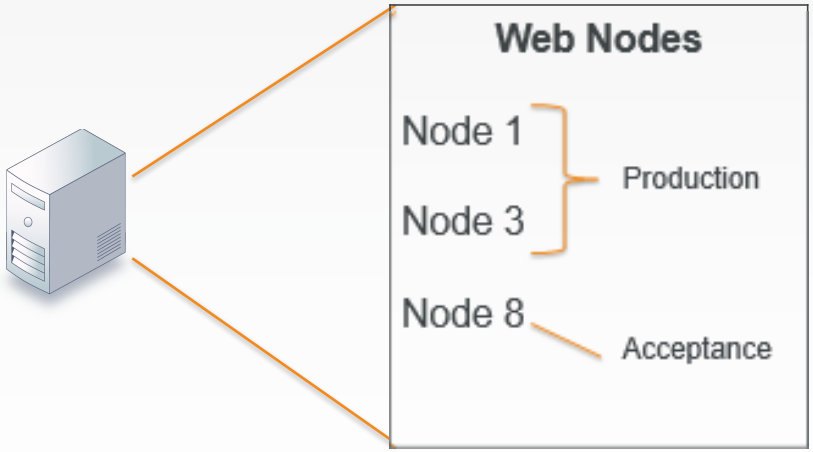
Objectives

After completing this module, you should be able to

- Create an environment
- Deploy a node to an environment
- Update a search query to be more exact

Environments

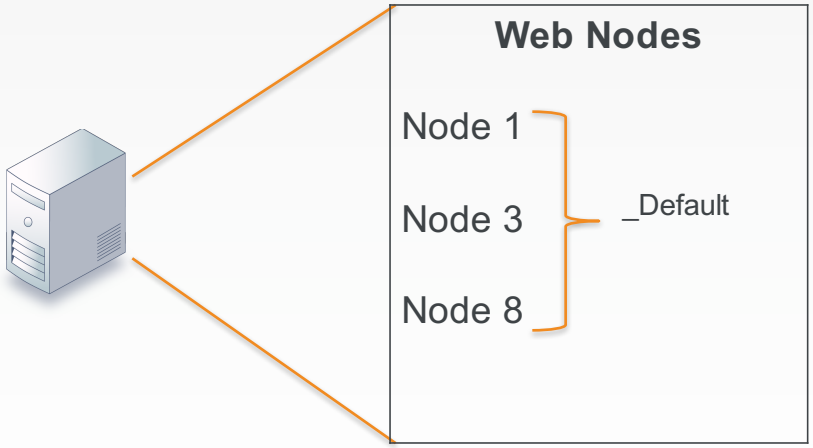
Environments can define different functions of nodes that live on the same system.

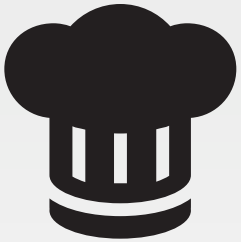


CHEF

Environments

Every organization or infrastructure starts with the _default environment.





Lab: Production

Let's create a reliable environment for our nodes.

Objective:

- ❑ Deploy Our Site to Production

Lab: Searching All of Our Nodes

```
$ knife search node "*:*"
```

```
3 items found

Node Name:   node1
Environment: _default
FQDN:        ip-172-31-29-218.ec2.internal
IP:          54.88.185.159
Run List:    role[web]
Roles:       web
Recipes:     apache::default, my-chef-client::default, apache::server, chef-
client::default, chef-client::service, chef-client::init_service
Platform:    centos 6.7
Tags:
```

Lab: Create an environments Directory

```
$ mkdir environments
```

Lab: Create the production.rb

chef-repo/environments/production.rb

```
name 'production'
description 'Where we run production code'

cookbook 'apache', '= 0.3.0'
override_attributes({
  "apache" => {
        "port" => 8181
      }
})
```

Lab: Upload the production.rb File

```
$ knife environment from file production.rb
```

```
Updated Environment production
```

```
$ knife environment show production
```

```
chef_type:           environment
cookbook_versions:
  apache: = 0.3.0
default_attributes:
description:         Where we run production code
json_class:          Chef::Environment
name:                production
override_attributes:
  apache:
    port: 8181
```

Lab: Set node1 to production environment

```
$ knife node environment set node1 production
```

```
node1:
 chef_environment: production
```

```
$ knife node show node1
```

```
Node Name:   node1
Environment: production
FQDN:        ip-172-31-8-68.ec2.internal
IP:          54.175.46.24
Run List:    role[web]
Roles:       web
Recipes:     apache, apache::default, apache::server
Platform:    centos 6.7
Tags:
```

Lab: Set More Nodes to Production

- Set node2 & node3's environment to production

Lab: Set node2's Environment to Production

```
$ knife node environment set node2 production
```

```
node2:
  chef_environment: production
```

```
$ knife node environment set node3 production
```

```
node3:
  chef_environment: production
```

Lab: Verify nodes are in Production Environment

```
$ knife search node "*:*"
```

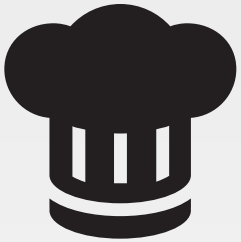
```
Node Name:   node1
Environment: production
FQDN:        ip-172-31-29-218.ec2.internal
IP:          54.88.185.159
Run List:    role[web]
Roles:       web
Recipes:     apache::default, my-chef-client::default, apache::server, chef-client::default,
chef-client::service, chef-client::init_service
Platform:    centos 6.7
Tags:

Node Name:   node2
Environment: production
FQDN:        ip-172-31-29-217.ec2.internal
```

Lab: Converge All Web Nodes

```
$ knife ssh "*:*" -x USER -P PWD "sudo chef-client"
```

```
ec2-54-88-169-195.compute-1.amazonaws.com Starting Chef Client, version 12.4.4
ec2-54-88-185-159.compute-1.amazonaws.com Starting Chef Client, version 12.4.4
ec2-54-84-233-7.compute-1.amazonaws.com   Starting Chef Client, version 12.4.4
ec2-54-88-169-195.compute-1.amazonaws.com resolving cookbooks for run list:
["haproxy", "my-chef-client"]
ec2-54-88-185-159.compute-1.amazonaws.com resolving cookbooks for run list:
["apache", "my-chef-client"]
ec2-54-88-169-195.compute-1.amazonaws.com Synchronizing Cookbooks:
ec2-54-88-169-195.compute-1.amazonaws.com   - windows
ec2-54-88-169-195.compute-1.amazonaws.com   - logrotate
ec2-54-88-169-195.compute-1.amazonaws.com   - chef-client
...
```



Production

Let's create a reliable environment for our nodes.

Objective:

- ✓ Deploy our site to Production

DISCUSSION

Acceptance Environment

There is a new mandate requirement for the **Public IP**, the **Public Hostname** and the **Environment** to be displayed on the homepage

This needs to be tested fully before going into production

Lab: Acceptance Environment

- ❑ Bump the Apache cookbook version and update the homepage accordingly
- ❑ Create an environment named "acceptance"
- ❑ Bootstrap a new web node into the acceptance environment using the new apache cookbook
- ❑ Run chef-client on all the nodes

Lab: Bump the cookbook version number

cookbooks/apache/metadata.rb

```
name              'apache'
maintainer        'The Authors'
maintainer_email  'you@example.com'
license           'all_rights'
description       'Installs/Configures apache'
long_description  'Installs/Configures apache'
version           '0.3.1'
```

Lab: Update index.html.erb

chef-repo/cookbooks/apache/templates/default/index.html.erb

```
<html>
  <body>
    <h1>Hello, world!</h1>
    <h2>ipaddress: <%= node['cloud']['public_ipv4'] %></h2>
    <h2>hostname: <%= node['cloud']['public_hostname'] %></h2>
    <h2>Environment: <%= "#{node.chef_environment}" %></h2>
</body>
</html>
```

Lab: Upload the Cookbook

```
$ cd cookbooks/apache
$ berks install
```

```
Resolving cookbook dependencies...
Fetching 'apache' from source at .
Fetching cookbook index from https://supermarket.chef.io...
Using apache (0.3.1) from source at .
```

```
$ berks upload
```

```
Uploaded apache (0.3.1) to: 'https://api.opscode.com:443/organizations/ORGNAME'
```

Lab: Acceptance Environment

- ✓ Bump the Apache cookbook version and update the homepage accordingly
- ❑ Create an environment named "acceptance"
- ❑ Bootstrap a new web node into the acceptance environment using the new apache cookbook
- ❑ Run chef-client on all the nodes

Lab: Create a New Environment File

chef-repo/environments/acceptance.rb

```
name 'acceptance'
description 'Where code and applications are tested'

cookbook 'apache', '= 0.3.1'
override_attributes({
  "apache" => {
        "port" => 8181
      }
})
```

Lab: Upload the .rb File

```
$ knife environment from file acceptance.rb
```

```
Updated Environment acceptance
```

```
$ knife environment list
```

```
_default
production
acceptance
```

Lab: Acceptance Environment

- ✓ Bump the Apache cookbook version and update the homepage accordingly
- ✓ Create an environment named "acceptance"
- ❑ Bootstrap a new web node into the acceptance environment using the new apache cookbook
- ❑ Run chef-client on all the nodes

Lab: Bootstrap a new node into the Acceptance Environment

```
$ knife bootstrap FQDN -x USER -P PWD --sudo -N node4 -r 'role[web]' -E acceptance
```

```
Connecting to 52.23.183.92
52.23.183.92 -----> Existing Chef installation detected
52.23.183.92 Starting first Chef Client run...
52.23.183.92 Starting Chef Client, version 12.4.4
52.23.183.92 resolving cookbooks for run list: ["apache", "my-chef-client"]
52.23.183.92 Synchronizing Cookbooks:
52.23.183.92   - logrotate
52.23.183.92   - chef_handler
52.23.183.92   - chef-client
52.23.183.92   - windows
52.23.183.92   - my-chef-client
...
```

Lab: Verify that the Environment Was Set

```
$ knife node show node4
```

```
Node Name:   node4
Environment: acceptance
FQDN:        ip-172-31-19-80.ec2.internal
IP:          52.23.183.92
Run List:    role[web]
Roles:       web
Recipes:     apache::default, my-chef-client::default, apache::server, chef-client::default, chef-client::service, chef-client::init_service
Platform:    centos 6.7
Tags:
```

Lab: Acceptance Environment

- ✓ Bump the Apache cookbook version and update the homepage accordingly
- ✓ Create an environment named "test"
- ✓ Bootstrap a new web node into the test environment using the new apache cookbook
- ❑ Run chef-client on all the nodes

Lab: Update Load balancing pool

```
$ knife ssh "role:loadbalancer" -x chef -P chef "sudo chef-client"
```

```
ec2-54-88-169-195.compute-1.amazonaws.com Starting Chef Client, version 12.4.4
ec2-54-88-169-195.compute-1.amazonaws.com resolving cookbooks for run list:
["haproxy", "my-chef-client"]
ec2-54-88-169-195.compute-1.amazonaws.com Synchronizing Cookbooks:
ec2-54-88-169-195.compute-1.amazonaws.com   - windows
ec2-54-88-169-195.compute-1.amazonaws.com   - logrotate
ec2-54-88-169-195.compute-1.amazonaws.com   - chef-client
ec2-54-88-169-195.compute-1.amazonaws.com   - chef_handler
ec2-54-88-169-195.compute-1.amazonaws.com   - my-chef-client
ec2-54-88-169-195.compute-1.amazonaws.com   - apache
```

Lab: Acceptance Environment

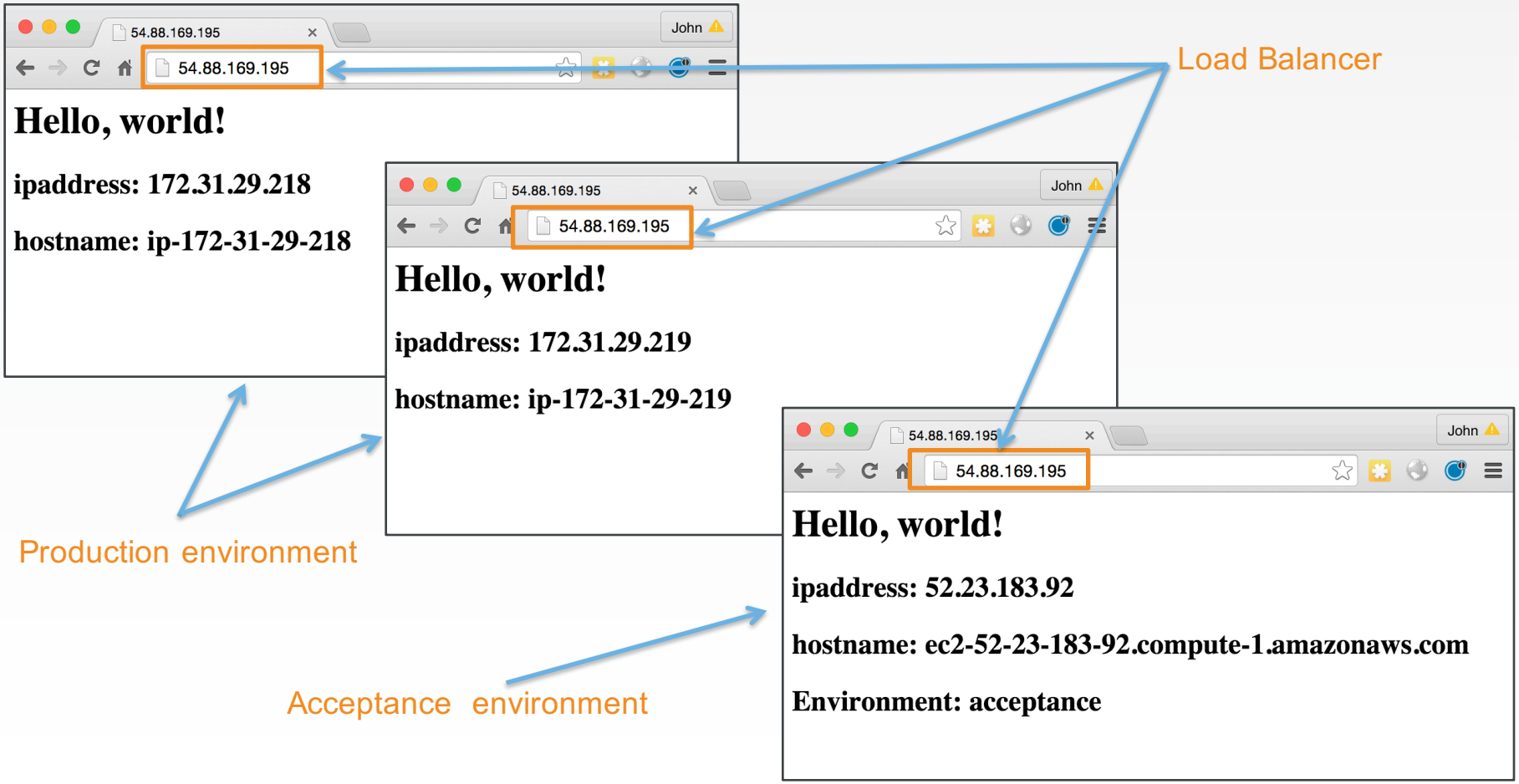
- ✓ Bump the Apache cookbook version and update the homepage accordingly
- ✓ Create an environment named "acceptance"
- ✓ Bootstrap a new web node into the acceptance environment using the new apache cookbook
- ✓ Run chef-client on all the nodes

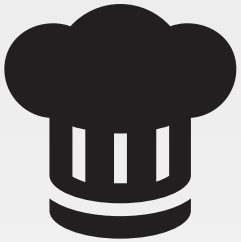
DISCUSSION

Separating Environments

Load balancer is distributing load across all environments

CHEF





Separating Environments

Objective:

- Use Search to separate out the environments

DISCUSSION

Balancing Nodes

Which cookbook handles balancing the requests between web nodes?

Which recipe within that cookbook sets up the request balancing between the two nodes?

Search Criteria

chef-repo/cookbooks/haproxy/recipes/default.rb

```
#
# Cookbook Name:: myhaproxy
# Recipe:: default
#
# Copyright (c) 2015 The Authors, All Rights Reserved.

webservers = search('node', 'recipes:apache\:\:default')

template '/etc/haproxy/haproxy.cfg' do
#...
```


CHEF

Lab: Modify the myhaproxy default.rb

chef-repo/cookbooks/haproxy/recipes/default.rb

```
#
# Cookbook Name:: myhaproxy
# Recipe:: default
#
# Copyright (c) 2015 The Authors, All Rights Reserved.

webservers = search('node', "role:web AND chef_environment:#{node.chef_environment}")

template '/etc/haproxy/haproxy.cfg' do
#...
```

Lab: Run 'berks install'

```
$ cd cookbooks/haproxy
$ berks install
```

```
Resolving cookbook dependencies...
Fetching 'apache' from source at ../apache
Fetching 'haproxy' from source at .
Using apache (0.3.1) from source at ../apache
Using apache2 (3.1.0)
Using haproxy (0.3.0) from source at .
```

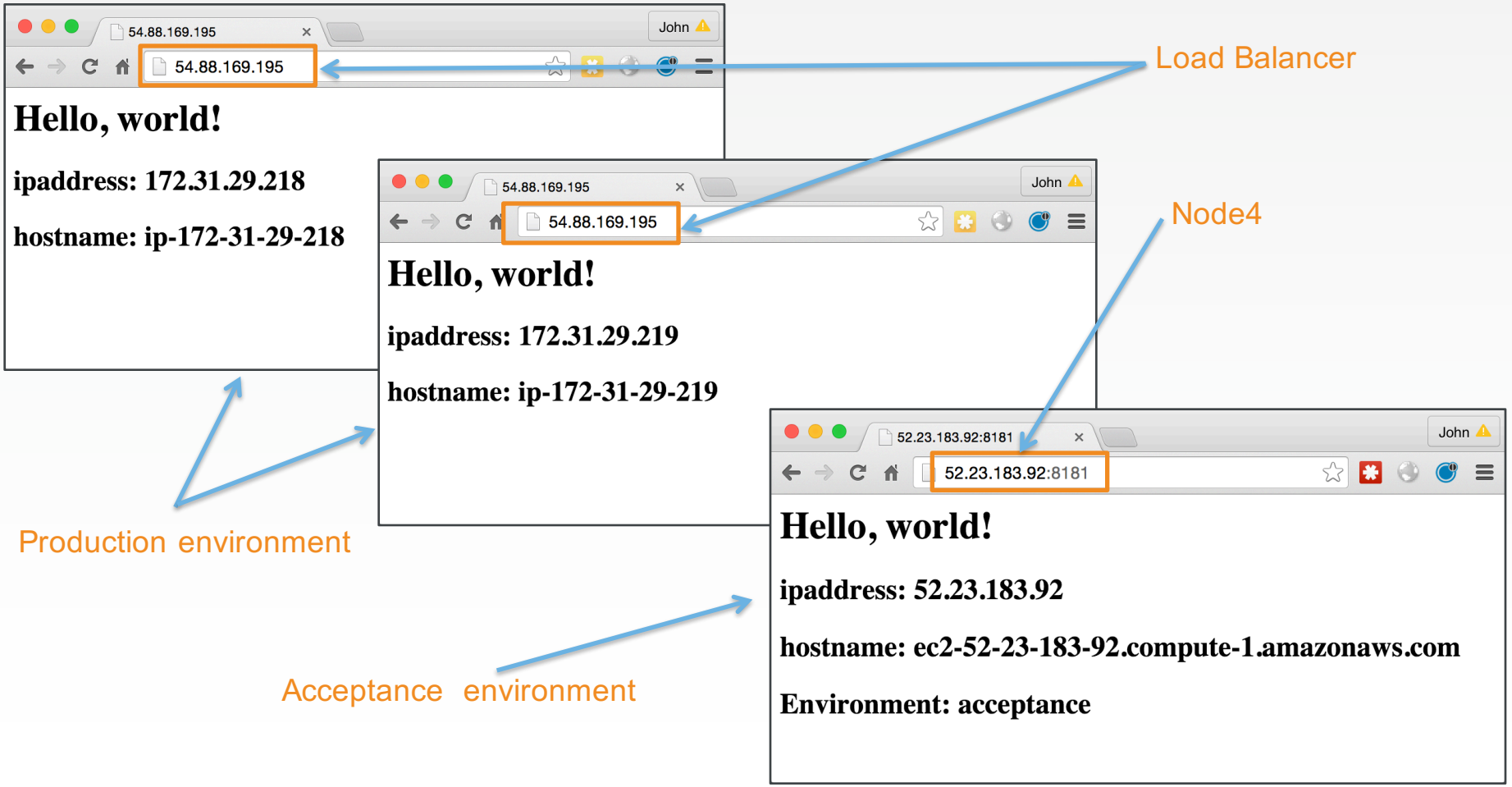
```
$ berks upload
```

```
Skipping apache (0.3.1) (frozen)
Uploaded haproxy (0.3.0) to: 'https://api.chef.io:443/organizations/ORGNAME'
```

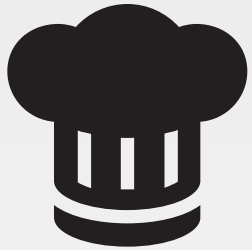
Lab: Run chef-client on all nodes

```
$ knife ssh "*:*" -x USERNAME -P PASSWORD "sudo chef-client"
```

```
...
ec2-54-88-169-195.compute-1.amazonaws.com   * directory[/var/log/chef] action create (up to date)
ec2-54-88-169-195.compute-1.amazonaws.com   * directory[/etc/chef] action create (up to date)
ec2-54-88-169-195.compute-1.amazonaws.com   * template[/etc/init.d/chef-client] action create (up to
date)
ec2-54-88-169-195.compute-1.amazonaws.com   * template[/etc/sysconfig/chef-client] action create (up to
date)
ec2-54-88-169-195.compute-1.amazonaws.com   * service[chef-client] action enable (up to date)
ec2-54-88-169-195.compute-1.amazonaws.com   * service[chef-client] action start (up to date)
ec2-54-88-169-195.compute-1.amazonaws.com
ec2-54-88-169-195.compute-1.amazonaws.com Running handlers:
ec2-54-88-169-195.compute-1.amazonaws.com Running handlers complete
ec2-54-88-169-195.compute-1.amazonaws.com Chef Client finished, 0/13 resources updated in 28.735845396
seconds
```



CHEF



Lab: Separate Environments

- ✓ Use Search to separate out the environments

DISCUSSION

A Brief Recap

We restricted the production environment to specific cookbook version.

We created an acceptance environment with no cookbook restrictions.

We set specific nodes to each of these environments.

We updated the haproxy's default recipe to include environment search criteria.

And we changed the version number in the myhaproxy metadata.rb file.

DISCUSSION

Discussion

What is the benefit of constraining cookbooks to a particular environment?

When defining search criteria what happens when you **AND** in a query?

What happens when you use an **OR** in a query?

DISCUSSION

Q&A

What questions can we help you answer?

CHEF™