Workstation Setup

Disable IE Enhanced Security Configuration

Configure Powershell (Enable QuickEdit Mode, Set Height buffer to 9999)

Install Chef http://www.getchef.com/chef/install

Or on Mac OS X or Linux: $ curl -L http://www.opscode.com/chef/install.sh | sudo bash

Create a new Chef account and download Starter Kit http://www.getchef.com

(Do not use an existing account. Generating a Starter Kit will re-create your user pem.)

```
PS\> cd c:\Users\You\chef-repo
PS\> dir
PS\> dir .chef
```

View c:\Users\You\chef-repo\.chef\knife.rb
```
current_dir = File.dirname(__FILE__)
log_level                :info
log_location             STDOUT
node_name                "USERNAME"
client_key               "#{current_dir}/USERNAME.pem"
validation_client_name "ORGNAME-validator"
validation_key           "#{current_dir}/ORGNAME-validator.pem"
chef_server_url     "https://api.opscode.com/organizations/ORGNAME"
cache_type               'BasicFile'
cache_options( :path =>"#{ENV['HOME']}/.chef/checksums" )
cookbook_path            ["#{current_dir}/../cookbooks"]

PS\> knife --version
PS\> knife help list
PS\> knife client list
```
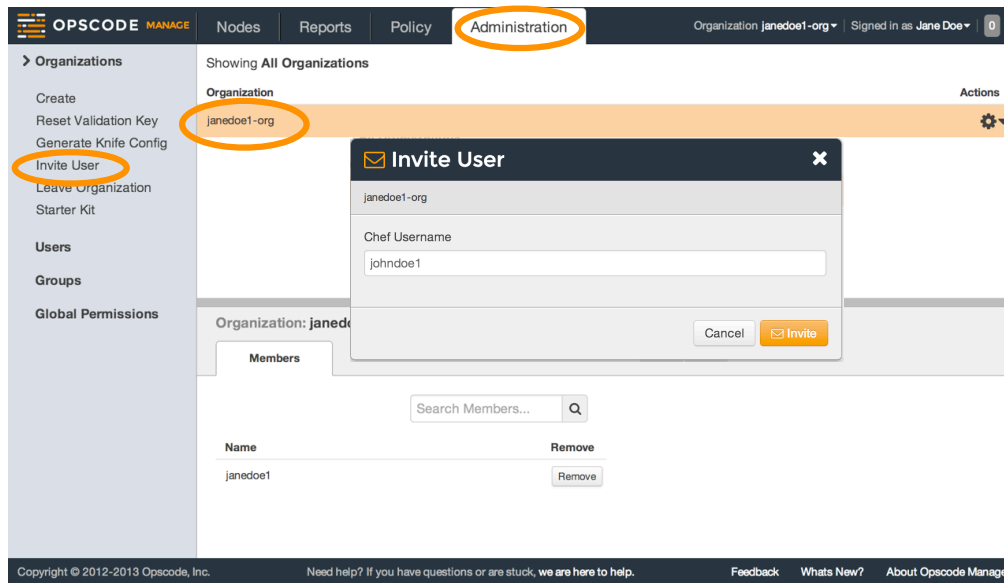
Organization Setup

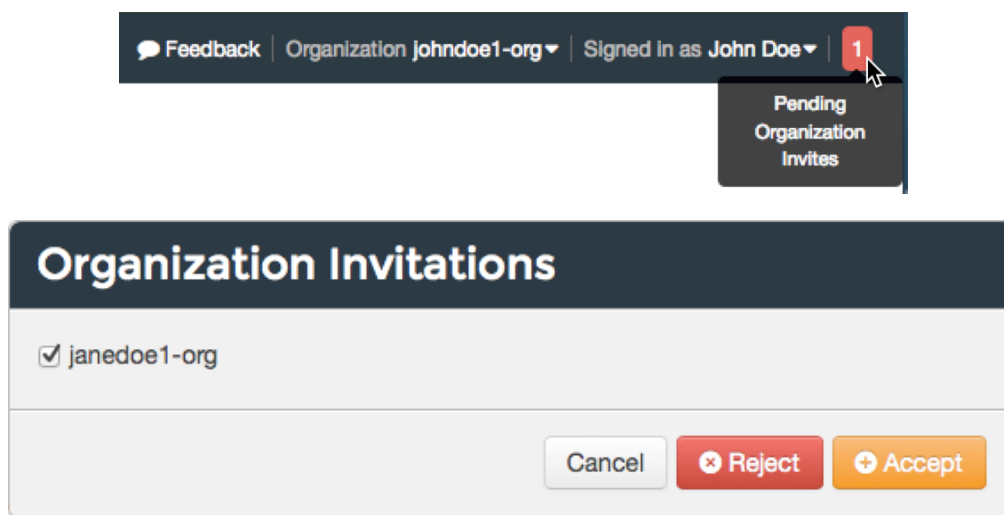Click the "Administration" tab,

Select the appropriate Organization

Click "Invite User" from the left menu

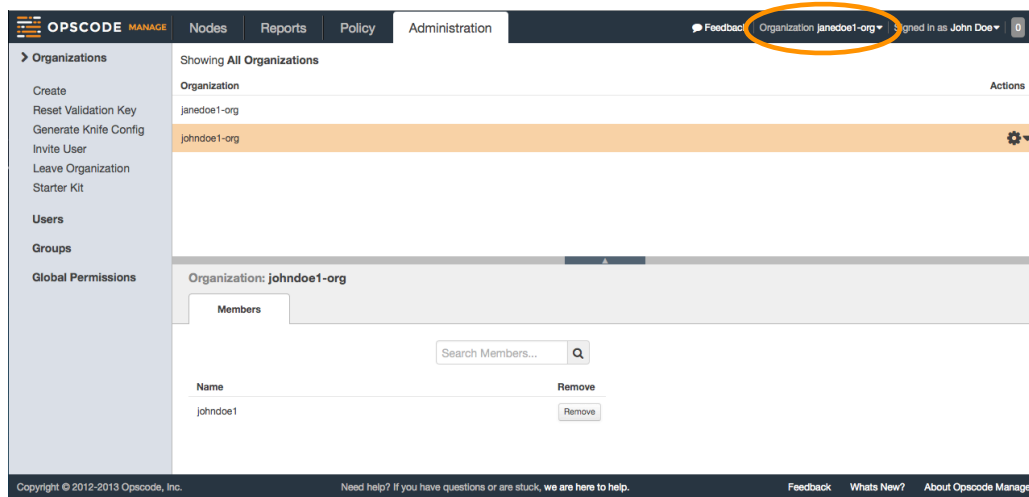Enter your classmate's 'Chef Username' and click Invite

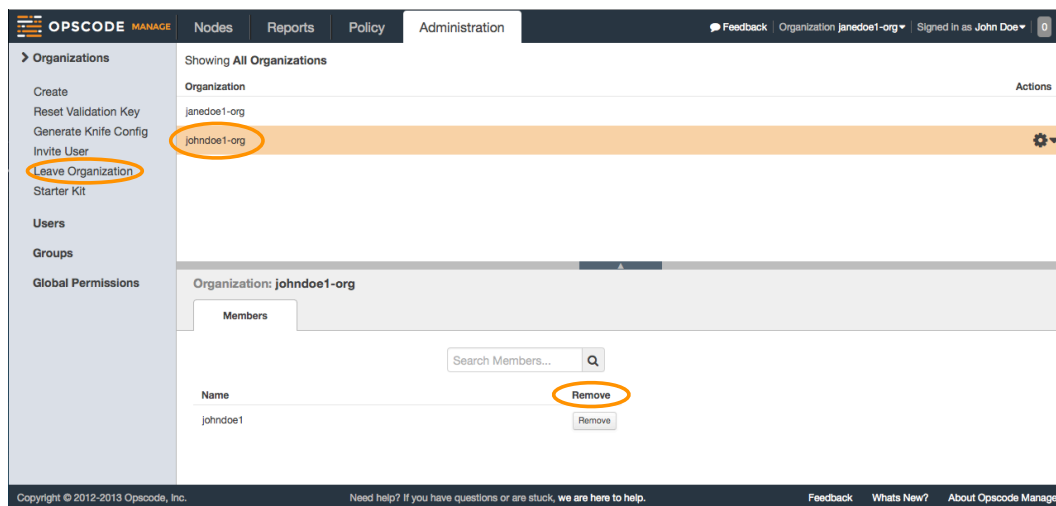Click the notification, select the Organization and click 'Accept'

Select your classmate's organization from the drop down list and peruse their org

Now either 'Leave Organization' you've been invited into, or remove your classmate from your organization
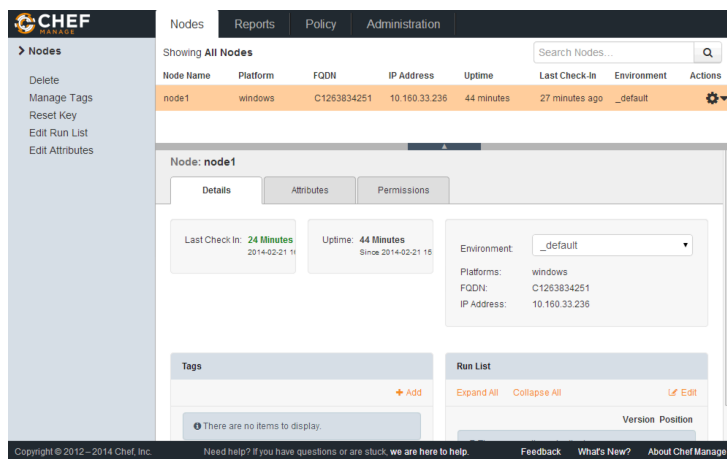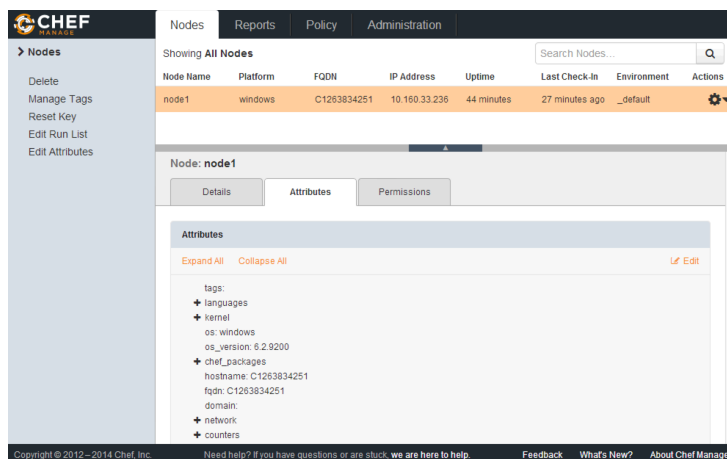
Test Node Setup

```
PS\> gem install knife-windows
PS\> knife bootstrap windows winrm <EXTERNAL_ADDRESS> -x chef -P
chef -N "node1"

rdp <EXTERNAL_ADDRESS>
Open Powershell as Administrator

remote@PS\> dir c:\chef
remote@PS\> gc c:\chef\client.rb
edit /etc/chef/client.rb
Set log_level to :info
remote@PS\> gc c:\chef\first-boot.json
```

View Node on Chef Server

Chef Resources and Recipes

```
PS\> knife cookbook create iis_demo
PS\> dir cookbooks\iis_demo
```

cookbooks\iis_demo\recipes\default.rb
```
powershell_script "Install IIS" do
  action :run
  code "add-windowsfeature Web-Server"
end

service "w3svc" do
  action [ :enable, :start ]
end

cookbook_file 'c:\inetpub\wwwroot\Default.htm' do
  source "Default.htm"
  rights :read, "Everyone"
end
```

cookbooks\iis_demo\files\default\Default.htm
```
<html>
<body>
  <h1>Hello, world!</h1>
</body>
</html>
```

```
PS\> knife cookbook upload iis_demo
PS\> knife node run_list add node1 'recipe[iis_demo]'
remote@PS\> chef-client
```

Introducing the Node Object

```
PS\> knife node list
PS\> knife client list
PS\> knife node show node1
remote@PS\> ohai | oh -Paging
PS\> knife node show node1 -l
PS\> knife node show node1 -Fj
PS\> knife node show node1 -a fqdn
PS\> knife search node "*:*" -a fqdn
```

Setting Node Attributes

Cookbooks\iis_demo\attributes\default.rb
```
default["iis_demo"]["indexfile"] = "Default1.htm"
```

cookbooks\iis_demo\files\default\Default1.html
```
<html>
 <body>
  <h1>Hello, world!</h1>
   <h2>This is Default1.htm</h2>
    <p>We configured this in the attributes file</p>
 </body>
</html>
```

cookbooks\iis_demo\recipes\default.rb
```
cookbook_file 'c:\inetpub\wwwroot\Default.htm' do
  source node["iis_demo"]["indexfile"]
  rights :read, "Everyone"
end
```

```
PS\> knife cookbook upload iis_demo
remote@PS\> chef-client
```

cookbooks\iis_demo\recipes\default.rb
node.default["iis_demo"]["indexfile"] = "Default2.htm"
```
cookbook_file 'c:\inetpub\wwwroot\Default.htm' do
  source node["iis_demo"]["indexfile"]
  rights :read, "Everyone"
end
```

cookbooks\iis_demo\files\default\Default2.htm

```
<html>
 <body>
  <h1>Hello, world!</h1>
   <h2>This is Default2.htm</h2>
    <p>We configured this in the recipe</p>
 </body>
</html>

PS\> knife cookbook upload iis_demo
remote@PS\> chef-client

PS\> knife node show node1 -l | more
```

Attributes, Templates, and Cookbook Dependencies

```
PS\> knife cookbook create datacenter


cookbooks\datacenter\attributes\default.rb
default["datacenter"]["location"] = "china"


PS\> knife cookbook upload datacenter


PS\> knife cookbook create hosts
```

cookbooks\hosts\recipes\default.rb
```
template "#{ENV['windir']}\\System32\\drivers\\etc\\hosts" do
  source "hosts.erb"
end
```

cookbooks\host\templates\default\hosts.erb
```
#This file is managed by the server <%= node["fqdn"] %>
<% case node["datacenter"]["location"] -%>
<% when "china" -%>
    0.0.0.0    nytimes.com
<% end -%>


PS\> knife cookbook upload hosts
PS\> knife node run_list add node1 'recipe[hosts]'
remote@PS\> chef-client
```

\cookbooks\hosts\metadata.rb
```
depends        'datacenter'


PS\> knife cookbook upload hosts
remote@PS\> chef-client


remote@PS\> gc C:\Windows\System32\drivers\etc\hosts
remote@PS\> gci C:\chef\backup\Windows\System32\drivers\etc
PS\> knife search node "*:*" -a datacenter
```

cookbooks\datacenter\attributes\default.rb
```
default["datacenter"]["location"] = "russia"
```

```
PS\> knife cookbook upload datacenter
remote@PS\> chef-client
remote@PS\> gc C:\Windows\System32\drivers\etc\hosts
PS\> knife node show node1 -a datacenter
```

Template Variables, Notifications, and Controlling Idempotency

\cookbooks\iis_demo\metadata.rb
```
name              'iis_demo'
maintainer        'YOUR_COMPANY_NAME'
maintainer_email "YOUR_EMAIL"
license           "All rights reserved"
description       "Installs/Configures hosts"
long_description IO.read(File.join(File.dirname(__FILE__),
'README.md'))
version           "0.2.0"
```

\cookbooks\iis_demo\attributes\default.rb
```
default["iis_demo"]["sites"]["clowns"] = { "port" => 80 }
default["iis_demo"]["sites"]["bears"] = { "port" => 81 }
```

cookbooks/iis_demo/recipes/default.rb
(See https://gist.github.com/ae9187fb8a5fcedc5ee2)

```
powershell_script "Install IIS" do
  code "add-windowsfeature Web-Server"
  action :run
end


service "w3svc" do
  action [:enable, :start ]
end


powershell_script "disable default site" do
    code 'get-website "Default Web Site*" | where {$_.state -ne
"Stopped"} | Stop-Website'
end


node['iis_demo']['sites'].each do |site_name, site_data|
    site_dir =
"#{ENV['SYSTEMDRIVE']}\\inetpub\\wwwroot\\#{site_name}"
    directory site_dir

    powershell_script "create app pool for #{site_name}" do
        code "New-WebAppPool #{site_name}"
        not_if "C:\\Windows\\System32\\inetsrv\\appcmd.exe list
apppool #{site_name}"
    end

    powershell_script "new website for #{site_name}" do
        code <<-EOH
          Import-Module WebAdministration
          if (-not(test-path IIS:\\Sites\\#{site_name})){
                $NewWebsiteParams = @{Name= '#{site_name}';Port=
#{site_data["port"]};PhysicalPath= '#{site_dir}';ApplicationPool= '#{site_name}'}
                New-Website @NewWebsiteParams
          }
          elseif ((Get-WebBinding -Name #{site_name}).bindingInformation -ne
'*:#{site_data["port"]}:') {
                $CurrentBinding = (Get-WebBinding -Name
#{site_name}).bindingInformation
                $BindingParameters = @{Name= '#{site_name}';Binding=
$CurrentBinding;PropertyName= 'Port';Value = #{site_data["port"]} }
                Set-WebBinding @BindingParameters
          }
```

```
        Get-Website -Name #{site_name} | Where {$_.state -like 'Stopped'} | Start-
Website
      EOH
    end


    template "#{site_dir}\\Default.htm" do
        source "Default.htm.erb"
        rights :read, "Everyone"
        variables(
            :site_name => site_name,
            :port => site_data['port']
            )
        notifies :restart, "service[w3svc]"
    end
end
```

cookbooks\iis_demo\templates\default\Default.htm.erb
```
<html>
    <body>
        <h2>We love <%= @site_name %></h2>
        <%= node["ipaddress"] %>:<%= @port %>
    </body>
</html>

PS\> knife cookbook upload iis_demo
remote@PS\> chef-client
```

Search

```
PS\> knife search node "*:*"
PS\> knife search node "ipaddress:10.*"
PS\> knife search node "*:*" -a ipaddress
PS\> knife search node "ipaddress:10.*" -a ipaddress
PS\> knife search node "ipaddress:10* AND platform:windows"
PS\> knife search node "ipaddress:[10.0.* TO 10.2.*]"
```

cookbooks\iis_demo\recipes\ip-logger.rb
```
search("node","platform:windows").each do |server|
   log "The Windows servers in your organization have the following
FQDN/IP Addresses:- #{server['fqdn']}/#{server['ipaddress']}"
end
```

```
PS\> knife cookbook upload iis_demo
PS\> knife node run_list add node1 'recipe[iis_demo::ip-logger]'
remote@PS \> chef-client
```

```
PS\> knife node run_list remove node1 'recipe[iis_demo::ip-
logger]'
```

Recipe Inclusion, Data Bags, and Search

```
PS\> md data_bags
PS\> md data_bags\users
PS\> knife data_bag create users
```

data_bags\users\bobo.json
```
{
  "id": "bobo",
  "comment": "Bobo T. Clown",
  "password": "iAm@Cl0wN!!!"
}
```

```
PS\> knife data_bag from file users bobo.json
```

data_bags\users\frank.json
```
{
  "id": "frank",
  "comment": "Frank Belson",
  "password": "sP3nC3r4Hire!"
}
```

Use knife to upload the data_bag item 'frank.json' (command hidden)

```
PS\> knife search users "*:*"
PS\> knife search users "comment:bobo" -a id
```
Create a data_bag called 'groups' (2 commands hidden)

data_bags\groups\clowns.json
```
{
  "id": "clowns",
  "members": [ "bobo", "frank" ]
}
```

Use knife to upload the data_bag item 'clowns.json' (command hidden)
Use knife to search the data bag 'groups' (command hidden)
Create a cookbook called 'users' (command hidden)
Edit the 'user' cookbook's default recipe and add the following

```
search(:users, "*:*").each do |user_data|
  user user_data["id"] do
    comment user_data["comment"]
    password user_data["password"]
  end
end
include_recipe "users::groups"
```

cookbooks\users\recipes\groups.rb

```
search(:groups, "*:*").each do |group_data|
  group group_data["id"] do
    members group_data["members"]
  end
end
```

Upload the 'users' cookbook (command hidden)

Add the 'users' cookbook's default receipt to node1's run list (command hidden)
Use rerun 'chef-client' (command hidden)

```
remote@PS\> net users
remote@PS\> net localgroup
```

Roles

```
PS\> md roles
```

Roles\webserver.rb
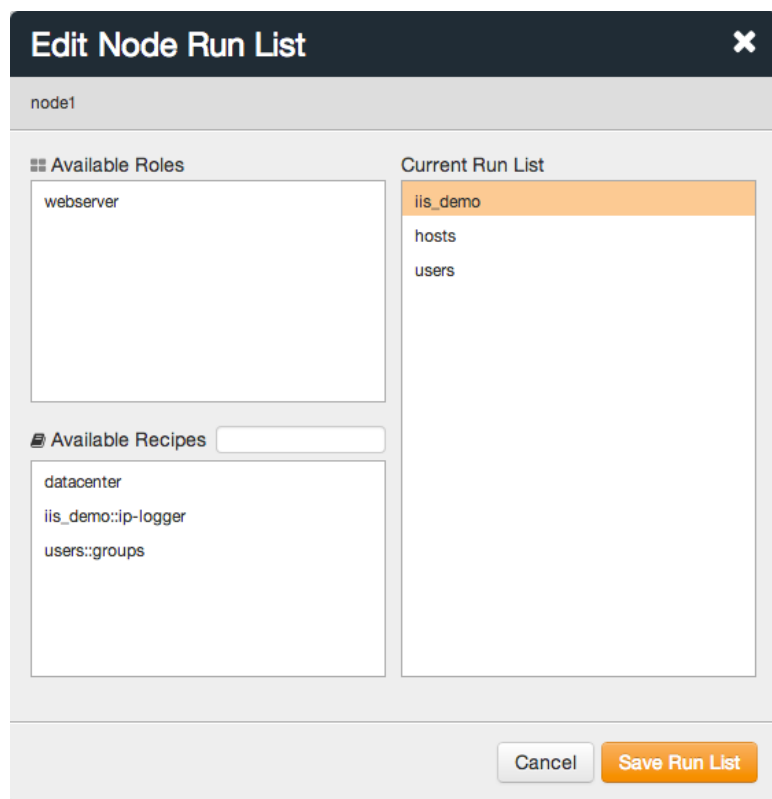
```
name "webserver"
description "Web Server"
run_list "recipe[iis_demo]"
default_attributes({
  "iis_demo " => {
    "sites" => {
      "admin" => {
        "port" => 8000
      }
    }
  }
})


PS\> knife role from file webserver.rb
PS\> knife role show webserver
```

Use knife to search for roles that have the iis_demo cookbook's default recipe in its run_list (command hidden)
Replace recipe[iis_demo] with role[webserver] in run list

```
remote@PS\> chef-client
PS\> knife search node "role:webserver" -a iis_demo.sites
```

roles\base.rb
```
name "base"
description "Base Server Role"
run_list "recipe[hosts]", "recipe[users]"
```

Use knife to upload this base role (command hidden)

roles\webserver.rb

```
name "webserver"
description "Web Server"
run_list "role[base]", "recipe[iis_demo]"
default_attributes({
  "iis_demo " => {
    "sites" => {
      "admin" => {
        "port" => 8000
      }
    }
  }
})
```

Use knife to upload this webserver role & rerun chef-client (commands hidden)

Environments
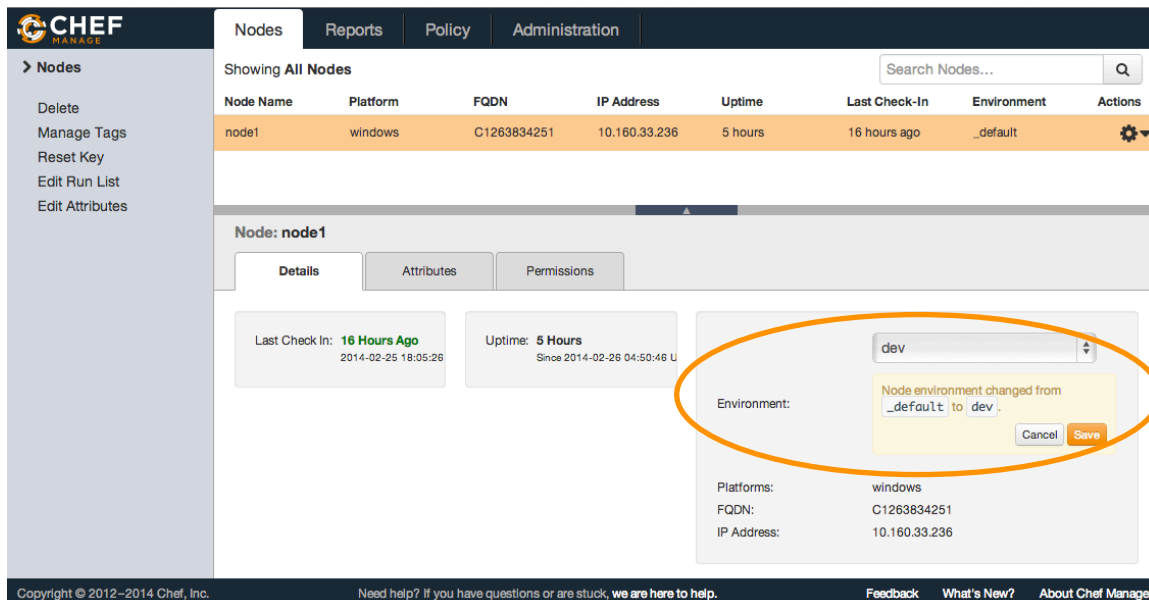
```
PS\> knife cookbook show iis_demo
PS\> knife environment list
PS\> mkdir environments
```

environments\dev.rb
```
name "dev"
description "For developers!"
cookbook "iis_demo", "= 0.2.0"
```

```
PS\> knife environment from file dev.rb
PS\> knife environment show dev
```
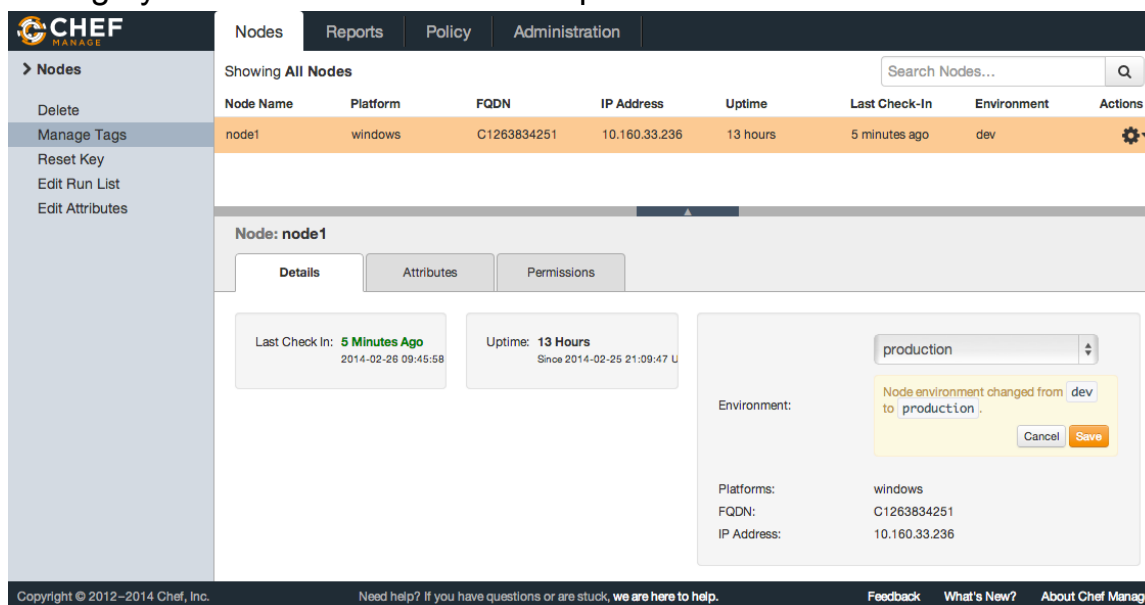
Change your node's environment to "dev"

Rerun chef-client (command hidden)

environments\production.rb

```
name "production"
 description "For Prods!"
 cookbook "iis_demo", "= 0.1.0"
 override_attributes({
    "datacenter" => {
      "location" => "china"
    }
 })
```

Use knife to create the production environment (command hidden)
Change your node's environment to "production"

Rerun chef-client (command hidden)

environments\production.rb
```
name "production"
 description "For Prods!"
 cookbook "iis_demo", "= 0.2.0"
 override_attributes({
    "datacenter" => {
      "location" => "china"
    }
 })
```

Use knife to create the production environment & rerun chef-client (commands hidden)

Using Community Cookbooks

```
PS\> knife cookbook site search chef-client
PS\> knife cookbook site show chef-client
PS\> knife cookbook site download chef-client
PS\> tar -zxvf chef-client*.tar.gz -C cookbooks/
```

cookbooks\chef-client\recipes\delete_validation.rb
```
unless chef_server?
  file Chef::Config[:validation_key] do
    action :delete
    backup false
    only_if { ::File.exists?(Chef::Config[:client_key]) }
  end
end
```

roles\base.rb
```
name "base"
description "Base Server Role"
run_list "recipe[chef-client::delete_validation]","recipe[hosts]",
"recipe[users]"
```

cookbooks\chef-client\recipes\default.rb
```
include_recipe "chef-client::service"
```

cookbooks\chef-client\recipes\service.rb

```
supported_init_styles = [
  'arch',
  'bluepill',
  'bsd',
  'daemontools',
  'init',
  'launchd',
  'runit',
  'smf',
  'upstart',
  'winsw'
]
init_style = node["chef_client"]["init_style"]

# Services moved to recipes
if supported_init_styles.include? init_style
  include_recipe "chef-client::#{init_style}_service"
else
  log "Could not determine service init style, manual intervention
required to start up the chef-client service."
end
```

Use knife to upload the 'chef-client' cookbook (command hidden)
Use knife to download the 'cron' cookbook (command hidden)
untar the 'cron' cookbook into the cookbooks directory (command hidden)
Use knife to upload the 'cron' cookbook (command hidden)
Use knife to upload the 'chef-client' cookbook (command hidden)
Use knife to download the 'logrotate' cookbook (command hidden)
untar the 'logrotate' cookbook into the cookbooks directory (command hidden)
Use knife to upload the 'logrotate' cookbook (command hidden)
Use knife to upload the 'chef-client' cookbook (command hidden)

`roles\base.rb`
```
name "base"
description "Base Server Role"
run_list "recipe[chef-client::delete_validation]",
"recipe[chef-client]", "recipe[hosts]", "recipe[users]"
```

Use knife to upload 'base' role and rerun chef-client (command hidden)
```
remote@PS\> get-service chef*
```