

Overview:

Chef enables you to automate your infrastructure. It provides a command line tool called knife to help you manage your configurations. Using the knife EC2 plugin you can manage your Amazon EC2 instances with Chef. knife EC2 makes it possible to create and bootstrap Amazon EC2 instances in just one line ? if you go through a few setup steps. Following are steps to setup your Chef installation and AWS configuration so that we can easily bootstrap new Amazon EC2 instances with Chef's knife

Following are the steps need to launch AWS instance.

A. Installation and Configuration of Knife Ec2 instance**1. Instaiing knife-ec2 instance:**

a. If you're using ChefDK, simply install the Gem:

```
$ chef gem install knife-ec2
```

b. If you're using bundler, simply add Chef and Knife EC2 to your Gemfile:

```
$ gem ?knife-ec2'
```

c. If you are not using bundler, you can install the gem manually from Rubygems:

```
$ gem install knife-ec2
```

In my setup I used ChefDK.

2. Add ruby's gem path to PATH variable to work knife-ec2 with AWS

```
$ export PATH=/root/.chefdk/gem/ruby/2.1.0/bin:$PATH
```

3. Add the AWS credentials of knife user to knife configuration file i.e ~/.chef/knife.rb.

```
????????????????????????????????
```

```
knife[:aws_access_key_id] = "user_key_ID"
```

```
knife[:aws_secret_access_key] = "User_secret_key"
```

```
????????????????????????????????
```

B. Prepare SSH access to Amazon EC2 Instance.**1. Configure Amazon Security Group**

As Amazon blocks all incoming traffic to EC2 instances by default. We'll need to open the SSH(22) port for knife to access a newly created instance. Also HTTPS(443) port to communicate launched instance's chef client with chefserver. Just login to the AWS management console and navigate to EC2 Services Compute Security Groups default group. Then add a rule for Type SSH and HTTPS with Source Anywhere and save the new

inbound rule

2. Generate Key Pair in AWS Console

To enable SSH access to Amazon EC2 instances you need to create a key pair. Amazon will install the public key of that key pair on every EC2 instance. knife will use the private key of that key pair to connect to your Amazon EC2 instances. Store the downloaded private key knife.pem in "~/.ssh/knife.pem" of ec2-user.

3. Prepare SSH configuration to avoid host key mismatch errors:

Create "/home/ec2-user/.ssh/config and add below content:

```
Host ec2*compute-1.amazonaws.com
StrictHostKeyChecking no
User ec2-user
IdentityFile /home/ec2-user/.ssh/knife.pem
```

C. Choose an AMI for your Amazon EC2 instances

We need to choose the right AMI for region, architecture and root storage. Note down the AMI ID (ami-XXXXXXXX) to use it with knife.

D. Create an EC2 instance using Chef knife:

Now, it's time to use knife to fire up and configure a new Amazon EC2 instance. Execute below command to create instance.

```
$sudo knife ec2 server create -r "recipe[dir]" -I ami-0396cd69 -f m3.large -S knife -i
/home/ec2-user/.ssh/knife.pem ?ssh-user ec2-user ?region us-east-1 -Z us-east-1b
```

Options:

-r is the run_list I want to associate with the newly created node. You can put any roles and recipes you like here
-I is the AMI ID
-f is the Amazon EC2 instance type
-S is the name you gave to the SSH key pair generated in the AWS management console
-i points to the private key file of that SSH key pair as downloaded when the key pair was created in the AWS management console
?ssh-user the official EC2 AMIs use ec2-user as the default user
?region us-east-1 If you want your instances to be deployed in any specific Amazon AWS region, add this parameter and the desired region
-Z us-east-1b is the availability zone within your region

NOTE:

If you did not give the ?r i.e run list with above mentioned command, then it throws the Exception below:

"EXCEPTIONS : NoMethodError Undefined method ?empty?' for nil:NilClass"

-
-
-
-