Chef Automate on the AWS Cloud

Quick Start Reference Deployment

AWS Quick Start Team

December 2015
Last updated: January 2018 (see revisions)

This guide is also available in HTML format at https://docs.aws.amazon.com/quickstart/latest/chef-server/.

Contents

About This Guide

This Quick Start reference deployment guide discusses configuration steps for deploying Chef Automate on the Amazon Web Services (AWS) Cloud. It also provides links for viewing and launching AWS CloudFormation templates that automate the deployment, and a walkthrough that demonstrates how you can manage Amazon Elastic Compute Cloud (Amazon EC2) instances with Chef Automate.

The guide is for IT infrastructure architects, administrators, and DevOps professionals who are planning to implement or extend their Chef workloads on the AWS Cloud.

> **AWS OpsWorks option** This Quick Start is for customers who want to run and manage their own Chef Automate infrastructure. However, we recommend that you also take a look at AWS OpsWorks, which is a configuration management service provided by AWS, to determine if it's more suitable for your needs. AWS OpsWorks helps you configure and operate applications of all types and sizes using Chef. You can define the application's architecture and the specification of each component, including package installation, software configuration, and resources such as storage. For more information, see the AWS OpsWorks User Guide.

Quick Starts are automated reference deployments for key workloads on the AWS Cloud. Each Quick Start launches, configures, and runs the AWS compute, network, storage, and other services required to deploy a specific workload on AWS, using AWS best practices for security and availability.

Overview

Chef Automate on AWS

Chef Automate is the highly scalable foundation of the Chef automation platform. You can use Chef Automate to create and manage dynamic infrastructure that runs on the AWS Cloud, or manage the servers in your on-premises data center. This Quick Start uses the official Chef Automate AMI from AWS Marketplace to make it easy to automate your infrastructure. You can launch the AMI from either AWS Marketplace or from this Quick Start to get instant access to the following features:

- Chef Automate, including Chef Server, Chef Analytics, Chef Management Console, and Chef Reporting
- Consolidated hourly billing for both Chef premium features and your infrastructure running on AWS
- The ability to manage up to 10 nodes for free.

This Quick Start is for customers who are planning to move to AWS, or are already running on AWS, and also want to deploy and manage their own Chef Automate infrastructure. In addition, the goal of this guide is to help you get started with Chef, even if you have no prior experience with the product.

This Quick Start automates the launch of the Chef Automate AMI, performs Chef Automate initial setup, creates the Chef administrative user account, and enables HTTPS access over the internet and within a virtual private cloud (VPC) in your AWS account. If you are new to Chef, you can choose to include an optional Chef workstation and managed node. You can follow the walkthrough included in this guide to learn how to configure the Chef workstation, create and upload a cookbook, and bootstrap an EC2 instance with Knife, which is a Chef command-line tool that helps you manage infrastructure components.

Quick Links

The links in this section are for your convenience. Before you launch the Quick Start, please review the architecture, configuration, network security, and other considerations discussed in this guide.

The default configuration deploys each server using the **t2.medium** instance type, but you can change the instance type and customize other settings. The supported instance types for Chef Automate are listed on the associated product page in the AWS Marketplace (see the next section).

Time to deploy: Approximately 35 minutes

Launch Quick Start

View template

Cost and Licenses

You are responsible for the cost of the AWS services used while running this Quick Start reference deployment. There is no additional cost for using the Quick Start. See the pricing pages for each AWS service you will be using in this Quick Start for full details.

Because this Quick Start uses AMIs from the AWS Marketplace, you must subscribe to Chef Automate from AWS Marketplace before you launch the Quick Start. The first ten nodes are free.

> **Important** If you try to launch the Quick Start without first subscribing to the Chef Automate AMI from the AWS Marketplace, the Chef Automate instance will not be created, and the stack creation will fail and roll back.

Architecture Overview

Deploying this Quick Start with the **default parameters** builds the following Chef Automate environment in the AWS Cloud.

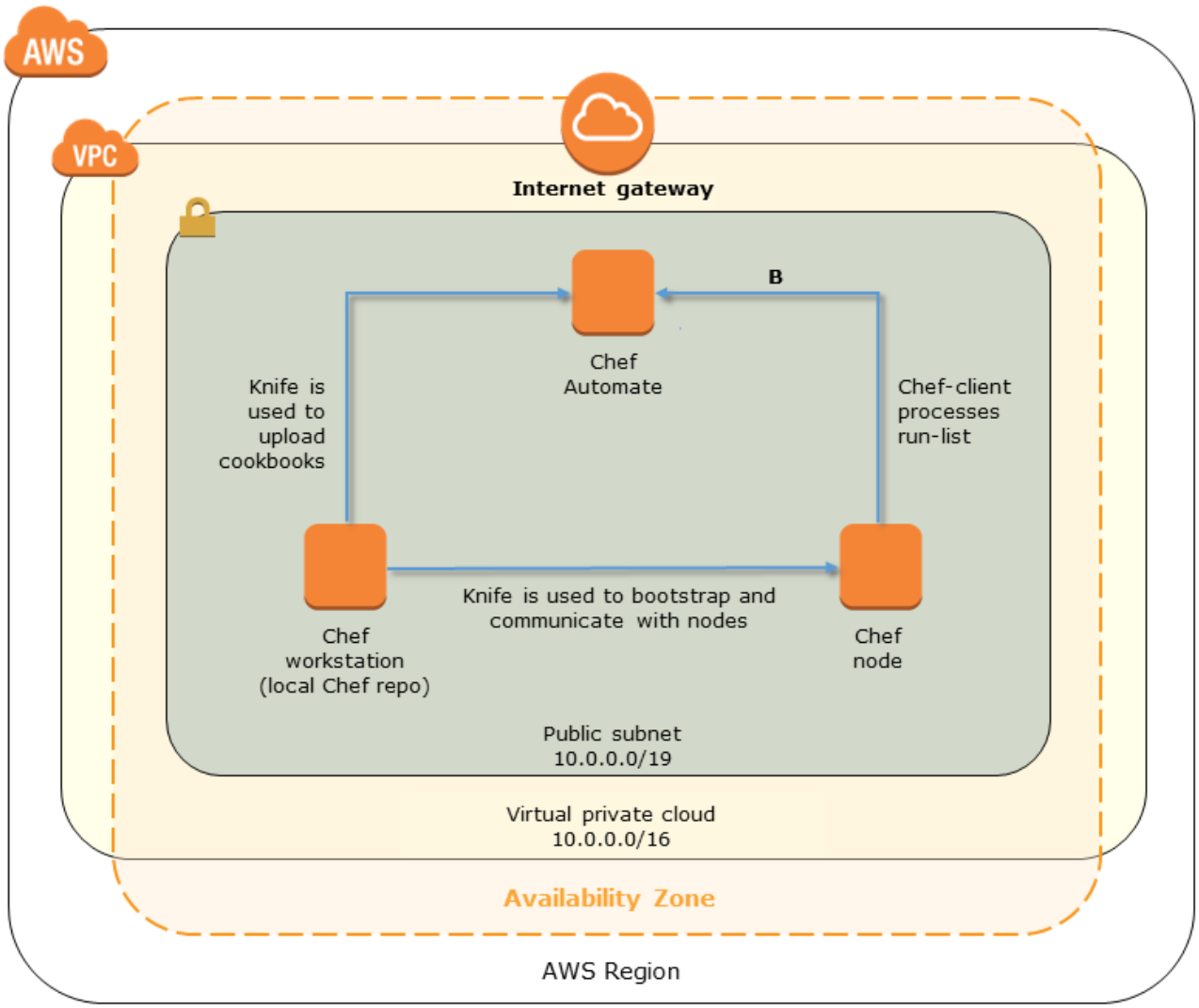

Figure 1: Quick Start architecture for Chef Automate on AWS

The resources deployed by this Quick Start and shown in Figure 1 are used as follows:

- A VPC is created in the region you choose when you launch the stack. A single public VPC subnet is created in the first Availability Zone.
- Chef Automate is deployed into the VPC subnet. An Elastic IP address is allocated and associated with the instance. During instance launch, Chef Automate is bootstrapped and the **marketplace-setup** command is run to configure the server. You provide the values required by **marketplace-setup** via AWS CloudFormation parameters at the time you launch the stack.
- A Chef workstation running on Microsoft Windows Server is deployed into the VPC subnet. This server has a public IP assigned and is accessible via Remote Desktop Protocol (RDP) over the internet. Both Git and the Chef Development Kit (Chef DK) are automatically installed on this machine via the AWS CloudFormation bootstrapping process. This Chef workstation is optional. You can use a workstation in your own on-premises environment as an alternative. This Quick Start uses Windows Server for the Chef workstation, because Windows provides a graphical user interface to the Chef Client tools and web-based Management Console, but you are free to use any supported operating system for your own Chef workstation.
- One Ubuntu Server node is deployed into the VPC subnet. After you deploy Chef on AWS, you can follow the walkthrough in this guide to configure a local Git repository (chef-repo) on the workstation, create and upload a cookbook to Chef Automate, and then bootstrap the node and run the cookbook to configure a basic web server. As with the Chef workstation, this node is optional. With Chef Automate running on AWS, you can manage other nodes that you deploy on AWS or nodes that are located in your on-premises data center.

AWS Services

The core AWS components used by this Quick Start include the following AWS services. (If you are new to AWS, see the Getting Started Resource Center.)

- Amazon VPC – The Amazon Virtual Private Cloud (Amazon VPC) service lets you provision a private, isolated section of the AWS Cloud where you can launch AWS services and other resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.
- Amazon EC2 – The Amazon Elastic Compute Cloud (Amazon EC2) service enables you to launch virtual machine instances with a variety of operating systems. You can choose

from existing Amazon Machine Images (AMIs) or import your own virtual machine images.

- Amazon Marketplace – AWS Marketplace is an online store where you can sell or buy software that runs on AWS. AWS Marketplace complements programs like Amazon DevPay and the AWS Partner Network (APN).

Use Cases and Benefits of Chef on AWS

Running your Chef Automate on AWS provides advantages in pricing, automation, resource management, and other areas, as explained in the following sections.

Usage-based Pricing and Consolidated Billing

Running your Chef Automate on Amazon EC2 gives you the ability to leverage hourly billing, and to pay only for what you use. With this Quick Start and the Marketplace AMI, you can choose the number of nodes you need support for, and the Chef Automate licensing costs will be included in your hourly rate for running the instance. If you're already running your infrastructure on AWS, or if you're in the process of migrating your workloads to AWS, launching the Chef Automate AMI through this Quick Start or through the AWS Marketplace allows you to consolidate your Chef and AWS costs into a single monthly bill.

Hybrid Deployments

If you're managing servers both on premises and in the cloud, Chef can fully support automation in your hybrid deployment. Chef Automate running on Amazon EC2 can be used with other EC2 instances and with servers you have running in your own data center. This Quick Start makes your Chef Automate instance internet-accessible, and supports automated hybrid deployments out of the box.

AWS Resource Management

Chef has the ability to manage AWS resources. If you want to use a single unified tool to manage your entire infrastructure, you can directly manage EC2 instances and leverage cookbooks to manage other resources, such as security groups, load balancers, Amazon Elastic Block Store (Amazon EBS) volumes, Elastic IP addresses, and tags. Take a look at Chef's aws cookbook on Chef Supermarket. This cookbook provides libraries, resources, and providers to configure and manage AWS components and offerings within the Amazon EC2 API.

Integration with AWS CloudFormation

With Chef, you can automate the deployment of your software applications on Amazon EC2 instances instead of manually writing and executing various scripts. By combining Chef with AWS CloudFormation, you can consistently deploy and configure your AWS resources, along with the software applications that run on top of AWS, all from a single AWS CloudFormation template. For a detailed walkthrough, see the Using Chef with AWS CloudFormation whitepaper provided by AWS.

Chef High Availability

Chef High Availability (HA) is a premium Chef feature that allows you to eliminate a single point of failure for your deployment. Chef HA includes a plugin for AWS that can move a virtual IP address and EBS volume from one Chef back end to another. For details, see High Availability on AWS in the Chef documentation.

Deployment Steps

The AWS CloudFormation template provided with this Quick Start bootstraps the AWS infrastructure and automates the deployment of Chef Automate on the AWS Cloud from scratch. Follow the step-by-step instructions in this section to set up your AWS account, customize the template, and deploy the software into your account.

What We'll Cover

The procedure for deploying the Chef Automate architecture on AWS consists of the following steps. For detailed instructions, follow the links for each step.

Step 1. Prepare an AWS account

Sign up for an AWS account, choosing a region, creating a key pair, and requesting increases for account limits, if necessary.

Step 2. Subscribe to the Chef Automate AMI

Subscribe to the Chef Automate AMI in AWS Marketplace.

Step 3. Launch the Quick Start

Launch the AWS CloudFormation template into your AWS account, specify parameter values, and create the stack.

Step 1. Prepare an AWS Account

1. If you don't already have an AWS account, create one at https://aws.amazon.com by following the on-screen instructions. Part of the sign-up process involves receiving a phone call and entering a PIN using the phone keypad.
2. Use the region selector in the navigation bar to choose the AWS Region where you want to deploy Chef Automate on AWS. For more information, see Regions and Availability Zones. Regions are dispersed and located in separate geographic areas. Each Region includes at least two Availability Zones that are isolated from one another but connected through low-latency links.

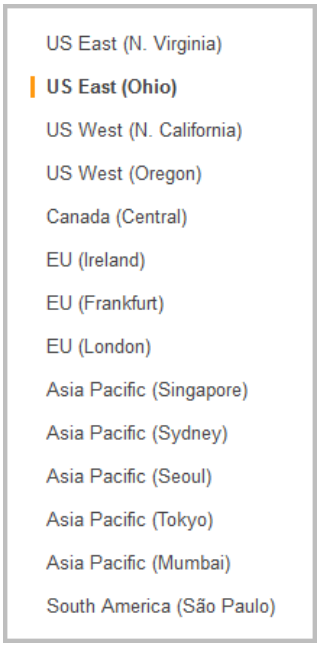

Figure 2: Choosing an AWS Region

> **Tip** Consider choosing a region closest to your data center or corporate network to reduce network latency between systems running on AWS and the systems and users on your corporate network.

3. Create a key pair in your preferred region. To do this, in the navigation pane of the Amazon EC2 console, choose **Key Pairs**, **Create Key Pair**, type a name, and then choose **Create**.

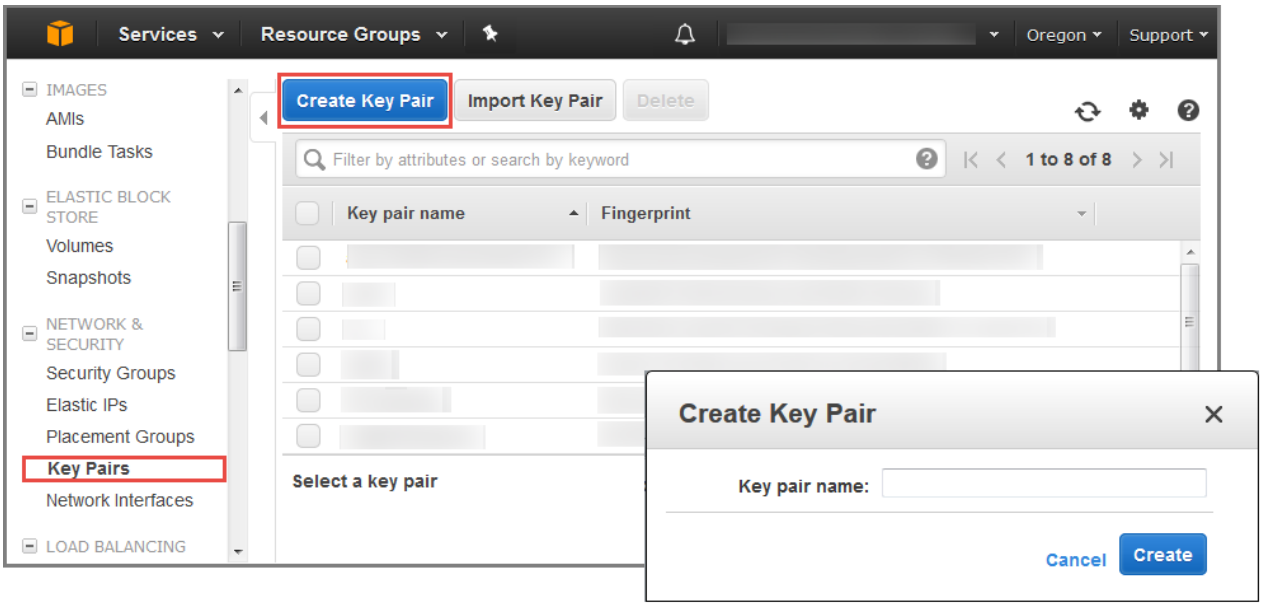

Figure 3: Creating a key pair

Amazon EC2 uses public-key cryptography to encrypt and decrypt login information. To log in to your instances, you must create a key pair. With Windows instances, the key pair is used to obtain the administrator password via the Amazon EC2 console and then log in using Remote Desktop Protocol (RDP), as explained in the Amazon EC2 User Guide. On Linux, the key pair is used to authenticate SSH login.

4. If necessary, request a service limit increase for the Amazon EC2 **t2.medium** instance type. To do this, in the AWS Support Center, choose **Create Case**, **Service Limit Increase**, **EC2 instances**, and then complete the fields in the limit increase form. The current default limit for this instance type is 20 instances.

   You might need to request an increase if you already have an existing deployment that uses this instance type, and you think you might exceed the default limit with this reference deployment. It might take a few days for the new service limit to become effective. For more information, see the Amazon EC2 User Guide.

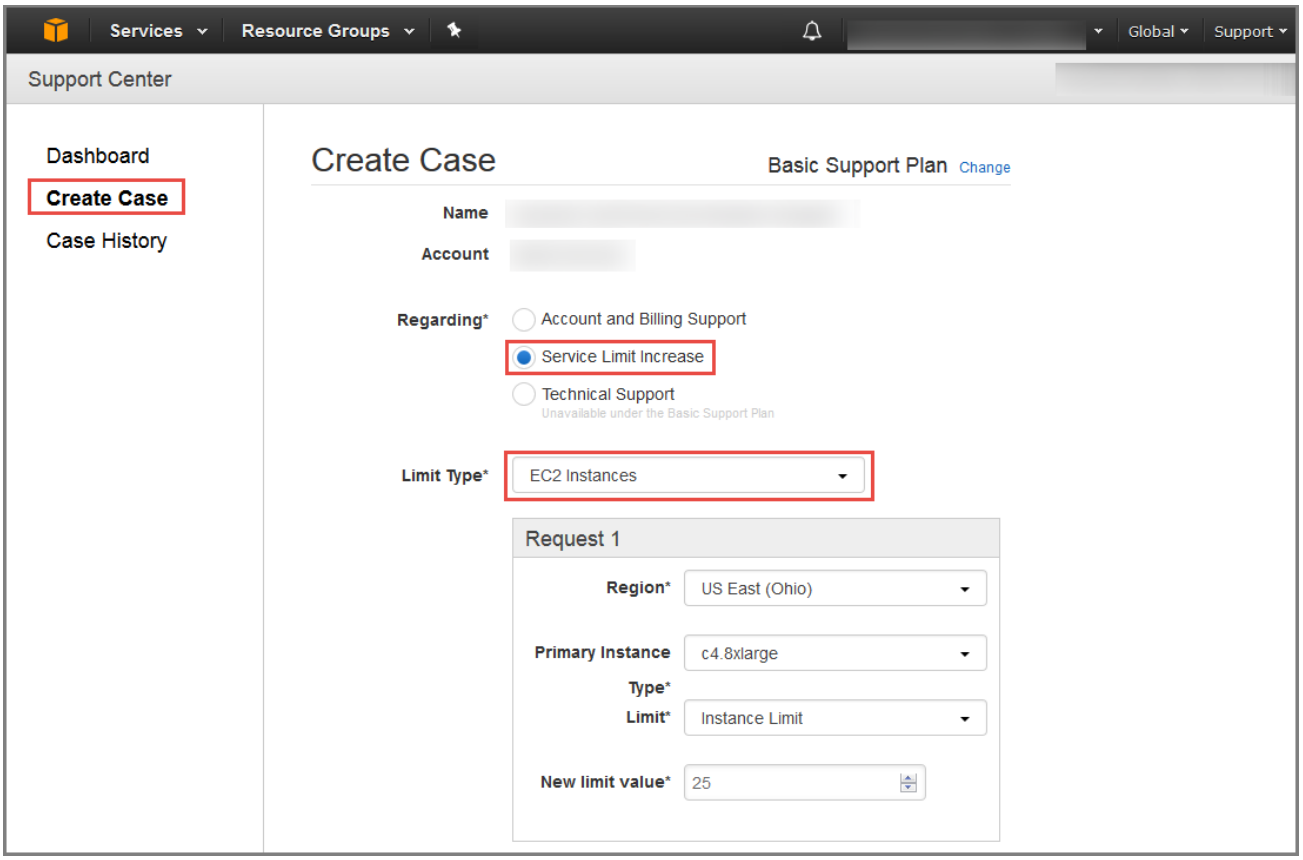

Figure 4: Requesting a service limit increase

Step 2. Subscribe to the Chef Automate AMI

1. Sign in to the AWS Marketplace at https://aws.amazon.com/marketplace.
2. Open the page for the Chef Automate AMI with 10 free nodes, and choose **Continue to Subscribe**.

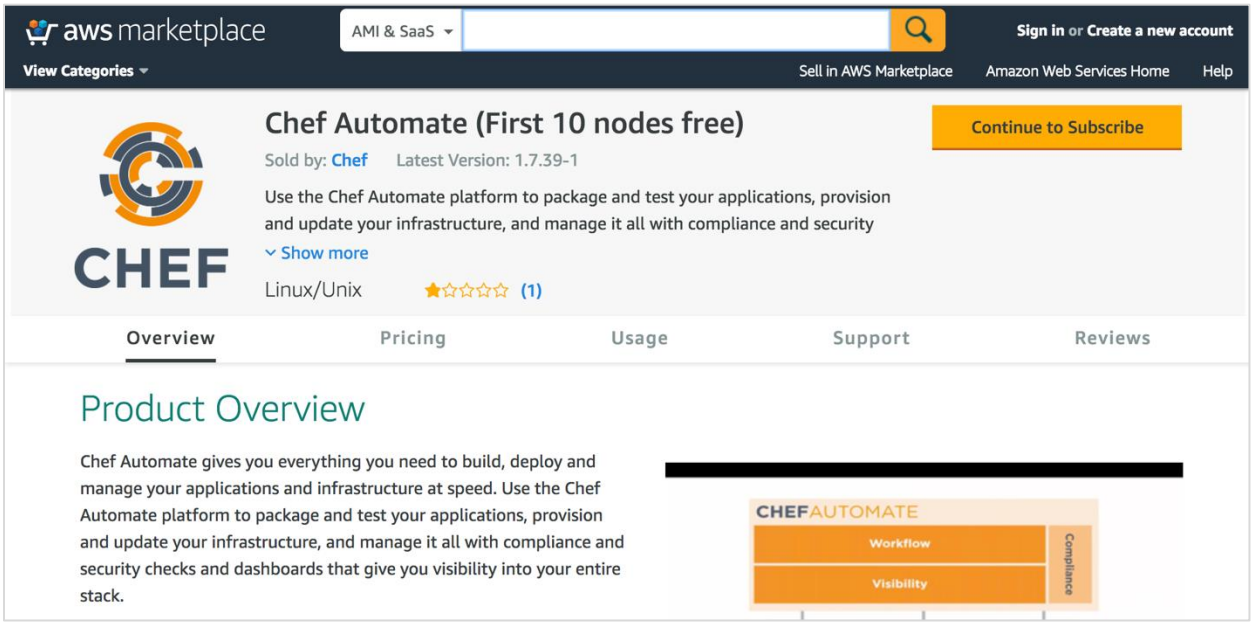

Figure 5: Selecting the Chef Automate AMI

3. Choose the **Manual Launch** tab, review the pricing details for the AMI, and then choose **Accept Software Terms**.

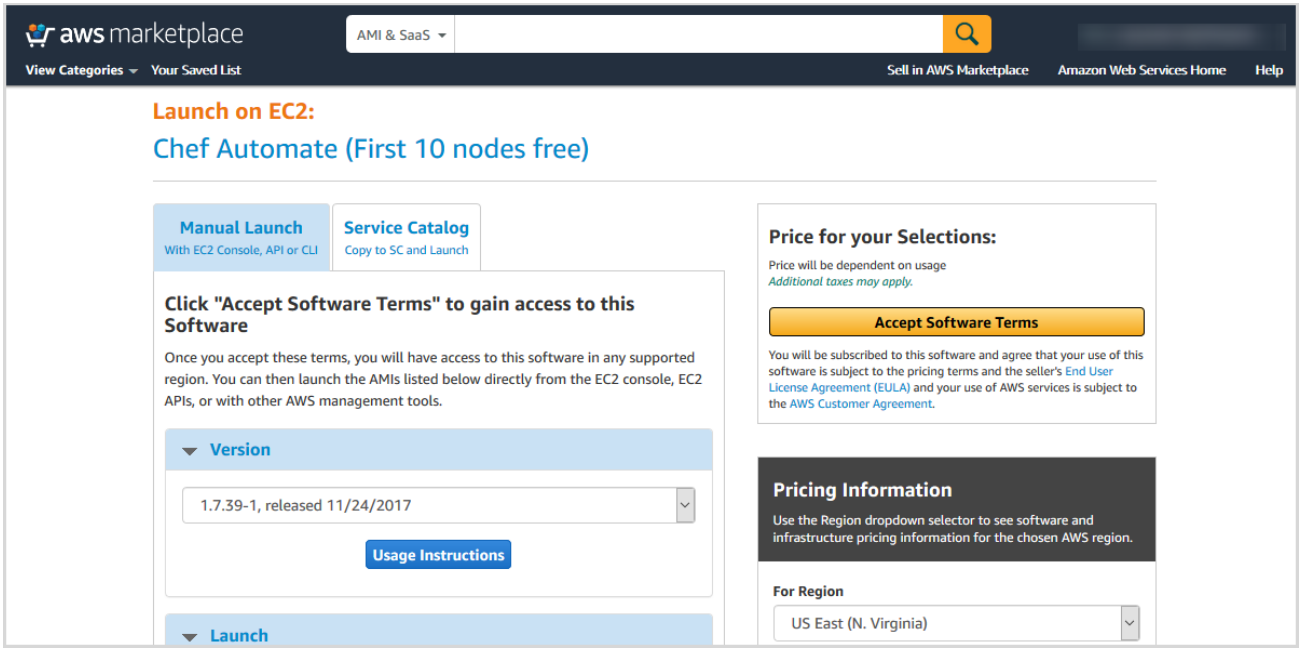

Figure 6: Accepting the terms of the Chef Automate AMI

After you accept the terms, you'll receive a notice that your subscription will be completed in a few moments.

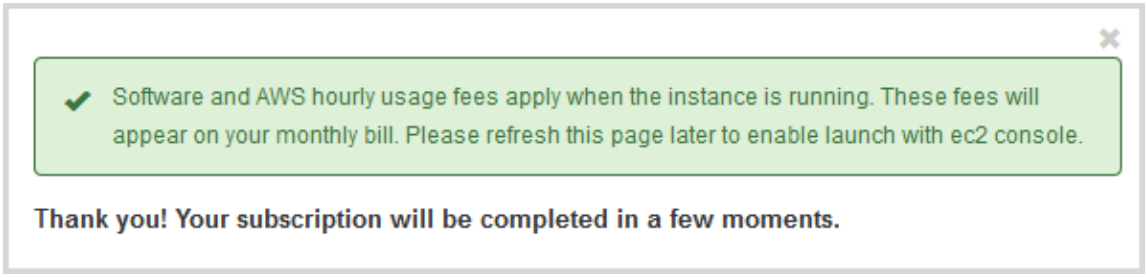

Figure 7: Chef Automate AMI subscription confirmation

3. Wait for email confirmation from AWS Marketplace notifying you that you are able to use this software in AWS. After you receive this confirmation, you are ready to launch the Chef stack.

Step 3. Launch the Chef Stack

This automated AWS CloudFormation template deploys Chef Automate into a new VPC. Please make sure that you've completed the previous step and have subscribed to the Chef Automate AMI in AWS Marketplace before launching the stack.

Launch stack

1. Launch the AWS CloudFormation template into your AWS account.

   The template is launched in the US East (Ohio) Region by default. You can change the region by using the region selector in the navigation bar.

   This stack takes approximately 35 minutes to create.

> **Note** You are responsible for the cost of the AWS services used while running this Quick Start reference deployment. There is no additional cost for using this Quick Start. See the pricing pages for each AWS service you will be using in this Quick Start for full details.

   You can also download the template to use it as a starting point for your own implementation.

2. On the **Select Template** page, keep the default URL for the AWS CloudFormation template, and then choose **Next**.

3. On the **Specify Details** page, review the parameters for the template. These are described in the following table.

   Provide values for the required parameters. For all other parameters, the template provides default settings that you can customize.

AWS Quick Start Configuration:

Parameter label (name)	Default	Description
Quick Start S3 Bucket Name (QSS3BucketName)	quickstart-reference	The S3 bucket you have created for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. The bucket name can include numbers, lowercase letters, uppercase letters, and hyphens, but should not start or end with a hyphen.
Quick Start S3 Key Prefix (QSS3KeyPrefix)	chef/chef-server/latest/	The S3 key name prefix used to simulate a folder for your copy of Quick Start assets, if you decide to customize or extend the Quick Start for your own use. This prefix can include numbers, lowercase letters, uppercase letters, hyphens, and forward slashes.

Network Configuration:

Parameter label (name)	Default	Description
VPC CIDR (VPCCIDR)	10.0.0.0/16	CIDR block for the VPC.
Enter the CIDR for your public subnet (PublicSubnetCIDR)	10.0.0.0/19	CIDR block for the private subnet.
CIDR for remote access (RemoteAdminCIDR)	*Requires input*	CIDR block or IP for SSH and RDP access to the Chef Automate software. We recommend that you set this value to a trusted IP range. For example, you might want to grant only your corporate network access to the software.

Chef Automate Configuration:

Parameter label (name)	Default	Description
Key Pair Name (KeyPairName)	*Requires input*	Public/private key pair, which allows you to connect securely to your instance after it launches. When you created an AWS account, this is the key pair you created in your preferred region.
Chef Automate Instance Type	t2.xlarge	EC2 instance type for master Chef Automate instances.

Parameter label (name)	Default	Description
(ChefAutomateInstance Type)		
Enable and Launch Demo Instances (IncludeDemoInstances)	true	Keep the default value to include a Chef workstation and demo node. To deploy Chef Automate only, set this parameter to **false**.
Chef Nodes (ChefNodes)	1	The number of nodes you want to support. The Chef Automate AMI allows you to manage up to 10 nodes for free.
Administrator password (AdminPassword)	*Requires input*	Password for the "chefadmin" user for signing in to the Chef workstation and Management Console. This must be a complex password that's at least 8 characters long.

4. On the **Options** page, you can specify tags (key-value pairs) for resources in your stack and set additional options. When you're done, choose **Next**.
5. On the **Review** page, review and confirm the template settings.
6. Choose **Create** to deploy the stack.
7. Monitor the status of the stack. When the status displays **CREATE_COMPLETE**, Chef Automate is ready.

Configuring Chef Automate

1. Use the Remote Desktop Protocol (RDP) client to connect to the Chef workstation. You can obtain the public DNS name or IP for the instance tagged as **ChefWorkstation** in the Amazon EC2 console. Use **chefadmin** for the user name and the password you provided when you launched the stack in step 3.
2. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
3. Copy the instance ID of the server labeled **ChefAutomate**.

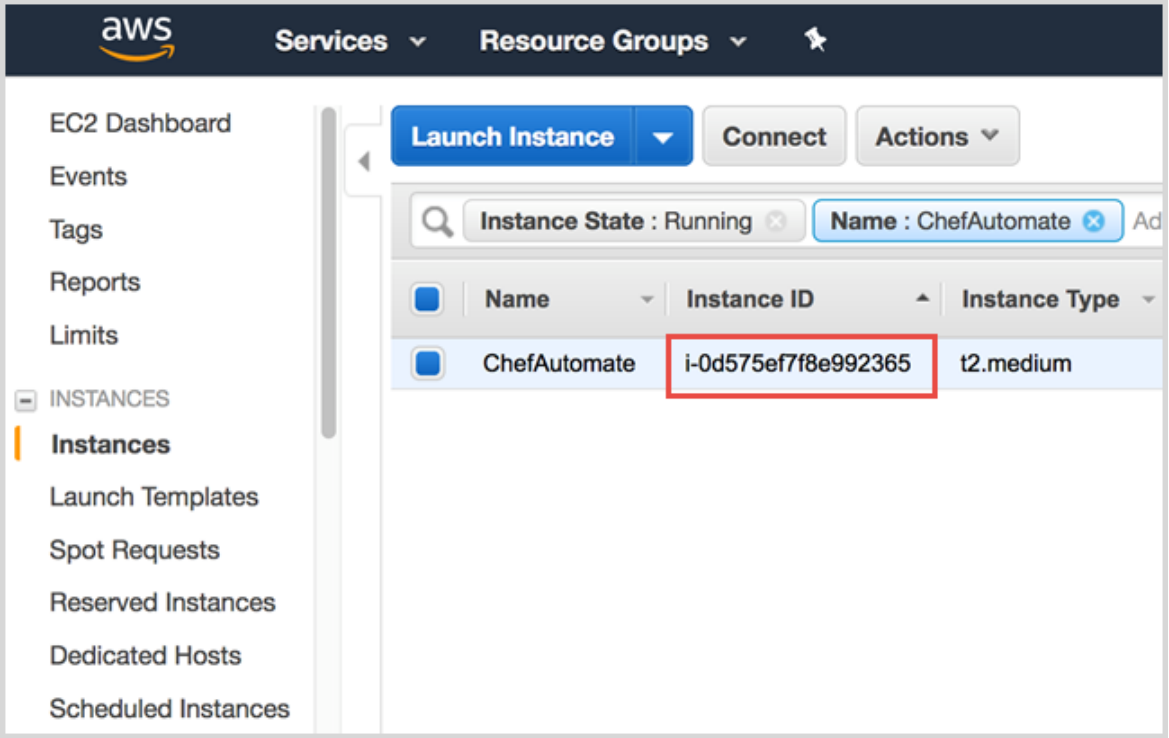

Figure 8: Copying the Chef Automate instance ID

4. Select the **ChefAutomate** instance, and in the **Description** pane, copy the public DNS name (e.g., `ec2-x-x-x-x.compute.amazonaws.com`).
5. Use the Google Chrome web browser installed on the workstation to navigate to the Chef Automate Management Console, using the URL https://*fqdn*/biscotti/setup, where *fqdn* is the DNS name for the Chef Automate server from the previous step.
6. Log in to the console by using the instance ID you copied earlier.

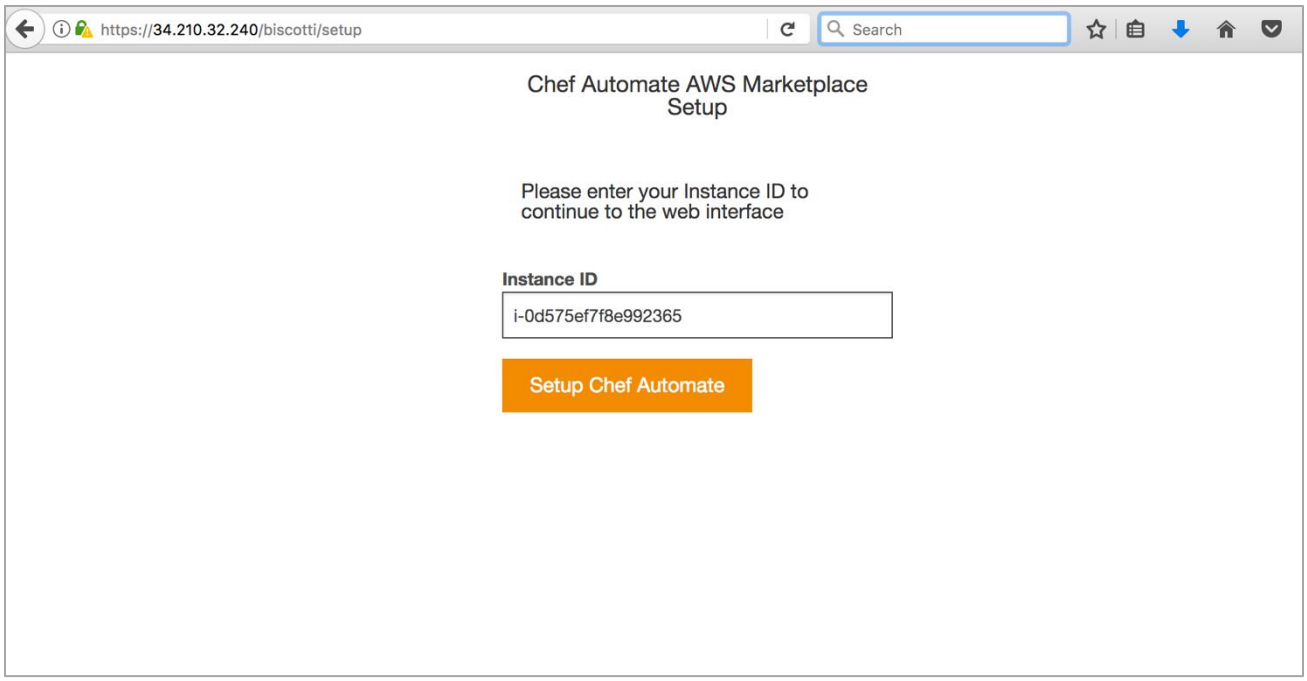

Figure 9: Pasting the Chef Automate instance ID

7. Fill out the form to create your Chef administrator account, read and accept the license terms, and then choose **Setup Chef Automate & Download Starter Kit**.

   The **starter_kit.zip** will be downloaded to the Chef workstation. Save this file to bootstrap your Chef nodes, as discussed in the following section.

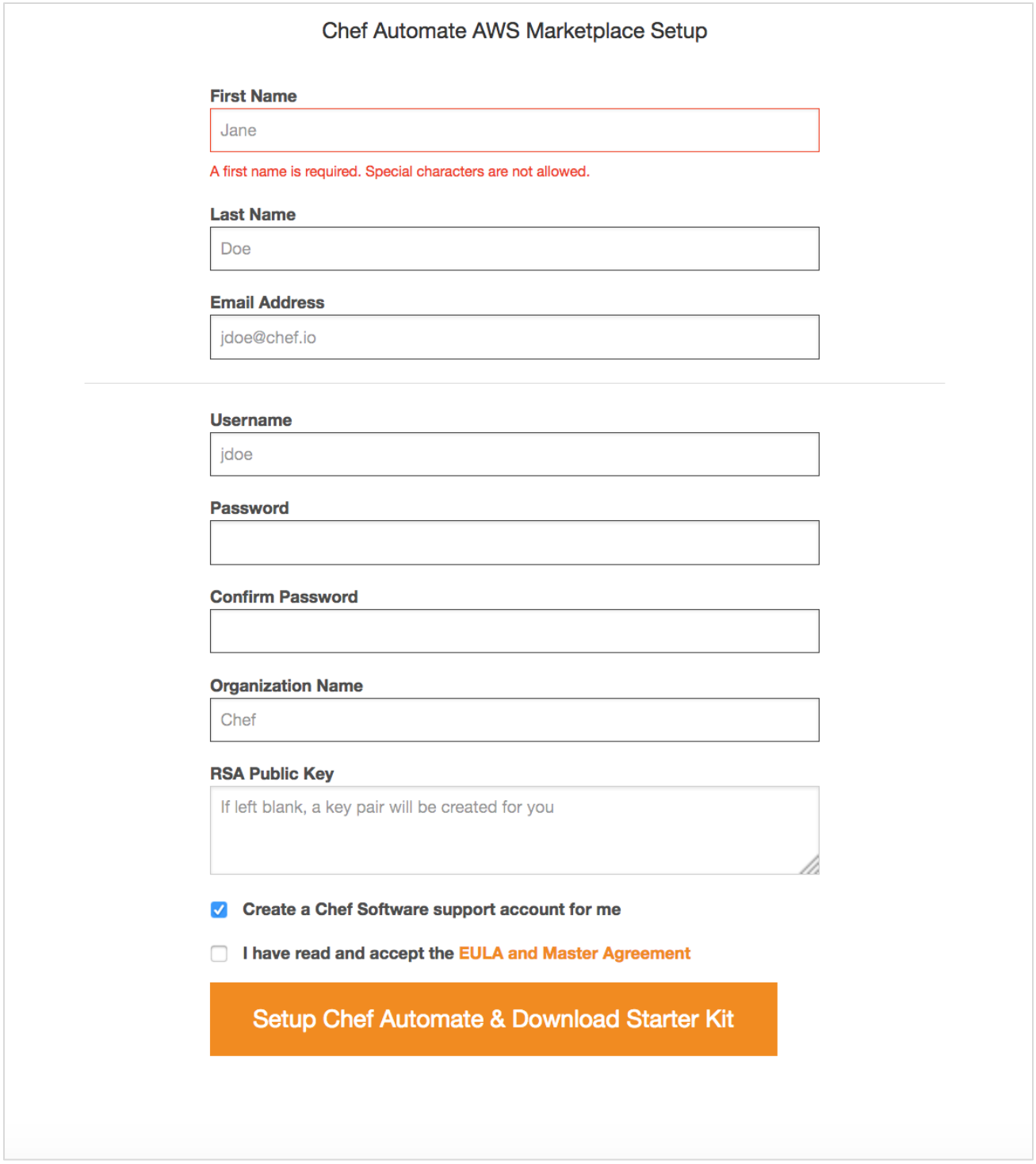

Figure 10: Creating your Chef Automate account

Configuring a Chef Node

If you chose the default setting for the **IncludeDemoInstances** parameter, you can follow this walkthrough to test your Chef Automate setup. In this section, we'll explain how to finalize the Chef workstation setup, create a cookbook, bootstrap a node, and verify that the configuration was applied.

Setting up the Workstation and chef-repo

1. On the desktop, open the context (right-click) menu for **Chef Development Kit**, and choose **Run as administrator**.

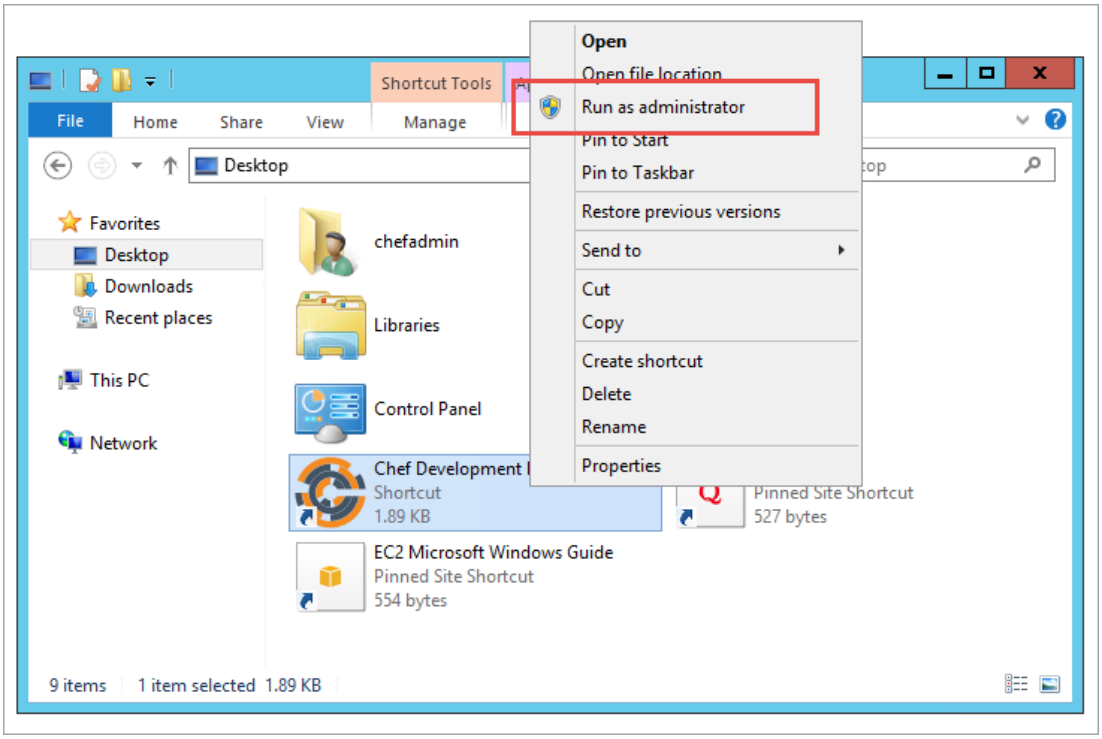

Figure 11: Launching the Chef Development Kit (ChefDK) from the desktop

This will bring up the ChefDK console, which runs via Windows PowerShell.

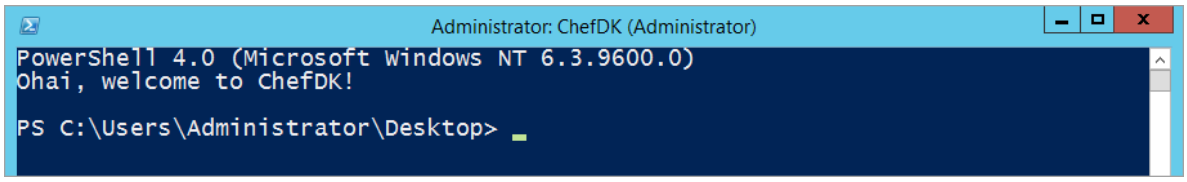

Figure 12: The ChefDK console

2. At the prompt, change to the root of drive C:

```
CD c:\
```

3. Run the following command to generate the chef-repo. This will be the Git repository used to store and version-control your cookbooks.

```
chef generate repo chef-repo
```

4. Run the following PowerShell commands to unzip the Chef Starter Kit. If you downloaded the archive to another location, update the source path as needed.

```
$source = 'C:\users\chefadmin\downloads\chef-starter.zip'
$dest = 'c:\chef-starter'
Expand-Archive -Path $source -DestinationPath $dest
```

5. Copy the **.chef** folder from the extracted Chef Starter Kit to the root of your chef-repo folder:

```
$source = 'c:\chef-starter\chef-repo\.chef'
Copy-Item -Path $source -Destination .\chef-repo -Recurse
```

6. Set the location of ChefDK console to the chef-repo folder:

```
CD .\chef-repo
```

   After this point, you must run all **knife** commands from this directory location. Do not switch (CD) to another location, or the **knife** commands will fail.

7. Run the **knife ssl fetch** command to add the self-signed certificate on Chef Automate to the trusted certs folder in your chef-repo. Replace the DNS name with the public DNS name of Chef Automate.

```
knife ssl fetch https://ec2-x-x-x-x.compute-1.amazonaws.com
```

   For production environments, you can replace the self-signed certificate with one that is issued from a trusted certification authority (CA).

Creating a Cookbook and Recipe

Now that the Chef workstation is fully configured, you're ready to create a cookbook.

1. In the ChefDK console, create a cookbook named `webserver` for a web server:

```
knife cookbook create webserver
```

2. To set up the cookbook, you can use sample files that are included with this Quick Start. Copy the provided recipe into your new cookbook:

```
$source = 'c:\cfn\downloads\default.rb'
$dest = 'c:\chef-repo\cookbooks\webserver\recipes'
Copy-Item -Path $source -Destination $dest
```

3. Copy a basic default web page into your `webserver` cookbook. This file will be used to show a "hello world" message to visitors who navigate to the web server after you've configured your node.

```
$source = 'c:\cfn\downloads\index.html'
$dest = 'c:\chef-repo\cookbooks\webserver\files'
Copy-Item -Path $source -Destination $dest
```

4. Open the recipe file in a text editor such as Notepad++ to review the code. The recipe defines a series of resources that will execute in order to configure your node as a web server.

```
10  #Run apt-get update
11
12  execute "apt-get update" do
13      command "apt-get update"
14  end
15
16  #Install apache
17
18  package "apache2" do
19      action :install
20  end
21
22  #Start apache service and enable service
23
24  service "apache2" do
25      action [:start, :enable]
26  end
27
28  cookbook_file "/var/www/html/index.html" do
29      source "index.html"
30      mode "0644"
31  end
```

Figure 13: The webserver recipe

Note the following about the code listed in Figure 13:

- Line 12 – The **execute** resource runs a single command—in this case, the **apt-get update** command. This command downloads package lists and updates them with the newest versions of packages and their dependencies.
- Line 18 – The **package** resource uses the appropriate package manager on the node to take the specified action on the package. In this case, we're installing the Apache 2.0 web server.
- Line 24 – The **service** resource starts the Apache service after it's installed, and then enables the service for automatic startup. The Linux distribution launched by this Quick Start already handles the automatic startup, but it's provided here for completeness, so you can use it as the basis for another server if necessary.
- Line 28 – The **cookbook_file** resource uses the index.html file you copied into the files folder of your `webserver` cookbook, and deploys it to the web server.

This is a very simple recipe that uses some of the common Chef resources. For more information about developing cookbooks, see the Additional Resources section.

5. Now that the cookbook is fully configured, upload it to Chef Automate:

```
knife cookbook upload webserver
```

Bootstrapping a Node

Next, you can use the **knife** command to bootstrap the node and execute the run-list for the node, which will contain the `webserver` cookbook.

1. Since **knife** will use SSH to communicate with the node, you'll need to provide your AWS private key. For example, if your private key is called **MyKey**, copy the MyKey.pem file to the `c:\chef-repo\.chef` folder. The .chef folder contains the .pem files for Chef Automate as well, and the .gitignore file in your chef-repo excludes the .pem file from Git commit operations.
2. You can now use the **knife bootstrap** command to bootstrap the node. Retrieve the private IP address for the Ubuntu Server tagged "NODE" in the Amazon EC2 console. Run the following command (replacing x.x.x.x with the private IP of your node, and mykey.pem with the name of your .pem file) to bootstrap the node and execute the run-list.

> **Note** The **knife bootstrap** command should be entered on one line in the ChefDK console. It's broken up here for readability.

```
knife bootstrap x.x.x.x
  --ssh-user ubuntu
  --sudo
  --identity-file ./.chef/mykey.pem
  --run-list webserver
```

After executing the command you'll see the node bootstrap, and the run-list will execute and configure the node as a web server. At that point, you can navigate to the node's IP address in a web browser, where you will see a "hello world" message confirming that the configuration has been applied successfully.

```
resolving cookbooks for run list: ["webserver"]
Synchronizing Cookbooks:
  - webserver (0.1.0)
Compiling Cookbooks...
Converging 4 resources
Recipe: webserver::default
  * execute[apt-get update] action run
    - execute apt-get update
  * apt_package[apache2] action install
    - install version 2.4.7-1ubuntu4.8 of package apache2
  * service[apache2] action start (up to date)
  * service[apache2] action enable (up to date)
  * cookbook_file[/var/www/html/index.html] action create
    - update content in file /var/www/html/index.html from 538f31
```

Figure 14: Reviewing the output when bootstrapping the node

You can configure the node to run chef-client on a regular basis to process future changes that you might make, such as modifying the cookbook, or customizing the run-list. To do so, schedule the chef-client to run at your desired interval via Cron. You can view the status of your nodes and workflows in the Chef Automate console.

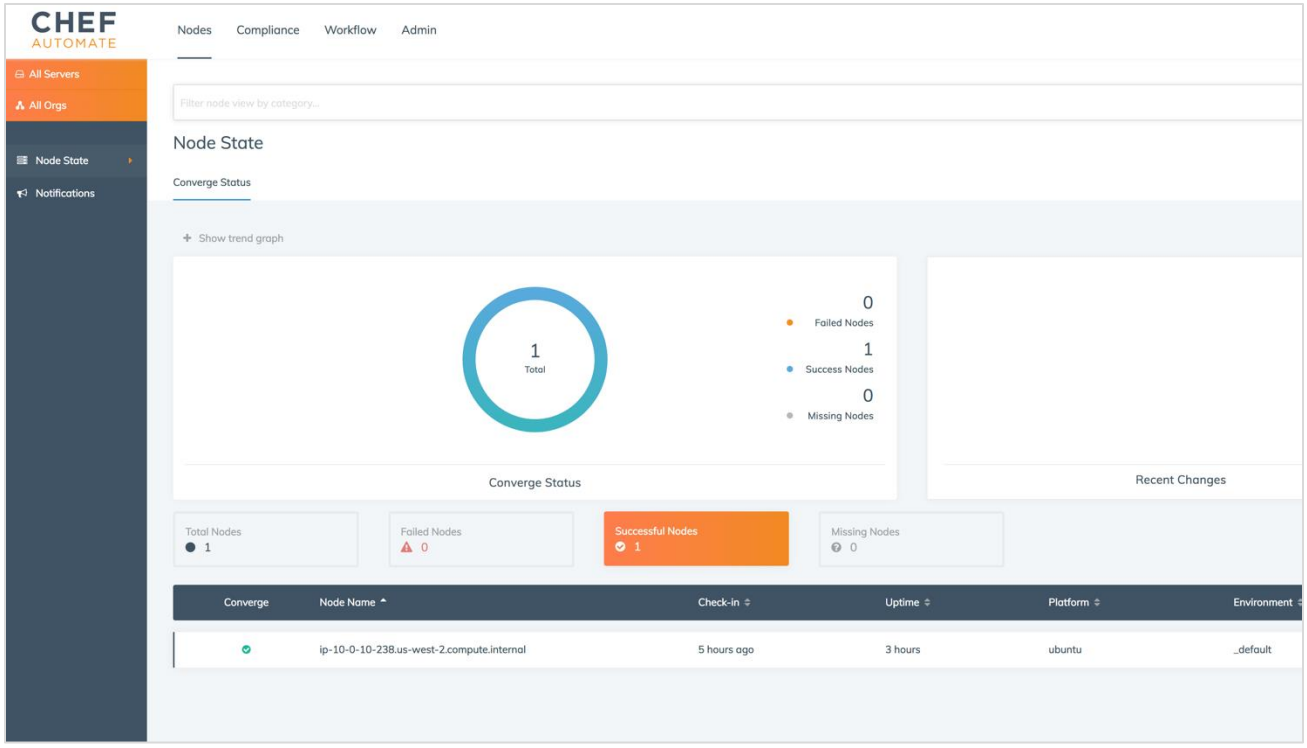

Figure 15: Viewing nodes in the Chef Automate console

Security

A *security group* acts as a firewall that controls the traffic for one or more instances. When you launch an instance, you associate one or more security groups with the instance. You add rules to each security group that allow traffic to or from its associated instances. You can modify the rules for a security group at any time. The new rules are automatically applied to all instances that are associated with the security group.

The security groups created and assigned to the individual instances as part of this solution are restricted as much as possible while allowing access to the various functions needed by Chef Automate. We recommend that you review security groups and further restrict access as needed once the deployment is up and running.

Support

Chef

The Chef Automate AMI includes a fast-response support channel available Monday through Friday, 6 AM to 6 PM Pacific Time, with experienced DevOps engineers. You can email the Chef Support team at aws@chef.io. See Chef Support Service Level Agreements for more details.

AWS

To extend the templates and scripts, to post your feedback, and to share your customizations with others, visit our GitHub repository.

Additional Resources

AWS services

- Amazon EC2
  https://aws.amazon.com/documentation/ec2/
- AWS CloudFormation
  https://aws.amazon.com/documentation/cloudformation/
- Amazon VPC
  https://aws.amazon.com/documentation/vpc/
- AWS OpsWorks
  https://aws.amazon.com/documentation/opsworks/

Chef resources

- Chef Automate AMI in AWS Marketplace
  https://aws.amazon.com/marketplace/pp/B01N813OWL
- Learn Chef
  https://learn.chef.io/
- Learn Chef Automate
  https://www.chef.io/automate/
- AWS Cookbook on Chef Supermarket
  https://supermarket.chef.io/cookbooks/aws

- Bootcamps at the AWS Loft
  https://www.chef.io/blog/events/?action=tribe_list&tribe_paged=1&tribe-bar-search=Loft
- Chef High Availability on AWS
  https://docs.chef.io/install_server_ha_aws.html
- Integrating AWS CloudFormation with Chef
  https://s3.amazonaws.com/cloudformation-examples/IntegratingAWSCloudFormationWithOpscodeChef.pdf
- Use Chef Cookbooks and Recipes with AWS OpsWorks
  http://docs.aws.amazon.com/opsworks/latest/userguide/workingcookbook.html

AWS Quick Start reference deployments

- AWS Quick Start home page
  https://aws.amazon.com/quickstart/

Send Us Feedback

You can visit our GitHub repository to download the templates and scripts for this Quick Start, to post your feedback, and to share your customizations with others.

Document Revisions

Date	Change	In sections
January 2018	Updated AMI subscription information per AWS Marketplace changes, updated parameters, revised configuration instructions for Chef Automate	Changes in templates and throughout guide
December 2015	Initial publication	-

Notices

This document is provided for informational purposes only. It represents AWS's current product offerings and practices as of the date of issue of this document, which are subject to change without notice. Customers are responsible for making their own independent assessment of the information in this document and any use of AWS's products or services, each of which is provided "as is" without warranty of any kind, whether express or implied. This document does not create any warranties, representations, contractual commitments, conditions or assurances from AWS, its affiliates, suppliers or licensors. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

The software included with this paper is licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with the License. A copy of the License is located at http://aws.amazon.com/apache2.0/ or in the "license" file accompanying this file. This code is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.