Chef is an open source technology developed by Opscode. Adam Jacob, co-founder of Opscode is known as the founder of Chef. This technology uses Ruby encoding to develop basic building blocks like recipe and cookbooks. Chef is used in infrastructure automation and helps in reducing manual and repetitive tasks for infrastructure management.

Chef have got its own convention for different building blocks, which are required to manage and automate infrastructure.

Why Chef?

Chef is a configuration management technology used to automate the infrastructure provisioning. It is developed on the basis of Ruby DSL language. It is used to streamline the task of configuration and managing the company’s server. It has the capability to get integrated with any of the cloud technology.

In DevOps, we use Chef to deploy and manage servers and applications in-house and on the cloud.

Features of Chef

Following are the most prominent features of Chef −

* Chef uses popular Ruby language to create a domain-specific language.
* Chef does not make assumptions on the current status of a node. It uses its mechanisms to get the current status of machine.
* Chef is ideal for deploying and managing the cloud server, storage, and software.

Advantages of Chef

Chef offers the following advantages −

* **Lower barrier for entry** − As Chef uses native Ruby language for configuration, a standard configuration language it can be easily picked up by anyone having some development experience.
* **Excellent integration with cloud** − Using the knife utility, it can be easily integrated with any of the cloud technologies. It is the best tool for an organization that wishes to distribute its infrastructure on multi-cloud environment.

Disadvantages of Chef

Some of the major drawbacks of Chef are as follows −

* One of the huge disadvantages of Chef is the way cookbooks are controlled. It needs constant babying so that people who are working should not mess up with others cookbooks.
* Only Chef solo is available.
* In the current situation, it is only a good fit for AWS cloud.
* It is not very easy to learn if the person is not familiar with Ruby.
* Documentation is still lacking.

Key Building Blocks of Chef

Recipe

It can be defined as a collection of attributes which are used to manage the infrastructure. These attributes which are present in the recipe are used to change the existing state or setting a particular infrastructure node. They are loaded during Chef client run and comparted with the existing attribute of the node (machine). It then gets to the status which is defined in the node resource of the recipe. It is the main workhorse of the cookbook.

Cookbook

A cookbook is a collection of recipes. They are the basic building blocks which get uploaded to Chef server. When Chef run takes place, it ensures that the recipes present inside it gets a given infrastructure to the desired state as listed in the recipe.

Resource

It is the basic component of a recipe used to manage the infrastructure with different kind of states. There can be multiple resources in a recipe, which will help in configuring and managing the infrastructure. For example −

* **package** − Manages the packages on a node
* **service** − Manages the services on a node
* **user** − Manages the users on the node
* **group** − Manages groups
* **template** − Manages the files with embedded Ruby template
* **cookbook\_file** − Transfers the files from the files subdirectory in the cookbook to a location on the node
* **file** − Manages the contents of a file on the node
* **directory** − Manages the directories on the node
* **execute** − Executes a command on the node
* **cron** − Edits an existing cron file on the node

Attribute

They are basically settings. They can be thought of as a key value pair of anything which one wants to use in the cookbook. There are several different kinds of attributes that can be applied, with a different level of precedence over the final settings that the node operates under.

File

It’s a subdirectory within the cookbook that contains any static file which will be placed on the nodes that uses the cookbooks. A recipe then can be declared as a resource that moves the files from that directory to the final node.

Templates

They are similar to files, but they are not static. Template files end with the .ebr extension, which means they contain embedded Ruby. They are mainly used to substitute an attribute value into the files to create the final file version that will be placed on the node.

Metadata.rb

It is used to manage the metadata about the package. This includes details like the name and details of the package. It also includes things such as dependency information that tells which cookbooks this cookbook needs to operate. This allows the Chef server to build the run-list of the node correctly and ensures that all of the pieces are transferred correctly.

Default Cookbook Structure

C:\chef\cookbooks\nginx>tree

Folder PATH listing for volume Local Disk

Volume serial number is BE8B-6427

C: ├───attributes

├───definitions

├───files

│ └───default

├───libraries

├───providers

├───recipes

├───resources

└───templates

└───default

Chef − Related Technologies

Following is the list of Chef related technologies.

Puppet

Puppet provides a standard way of delivering and operating software, no matter where it runs. It is an automated administrative engine for Linux, Unix, and Windows system that performs administrative tasks based on centralized specification.

The primary **features of Puppet** are as follows −

* Implementing new systems with a uniform configuration.
* Updating the systems and upgrading the security and software packages.
* Incorporating new features and adding dexterous capabilities.
* Customizing configurations for ensuring the availability of data sources.
* Optimizing the available resources and minimizing the cost.
* Simplifying the roles and enabling the team to focus on core and productive issues.
* Getting a bird’s eye view of the available infrastructure.

Ansible

Ansible is a radically simple IT automation platform that makes your applications and systems easier to deploy. Avoid writing scripts or custom code to deploy and update your applications — automate in a language that approaches plain English, using SSH, with no agents to install on remote systems.

The primary **features of Ansible** are as follows −

* Simple and easy to learn
* Written in Python
* Agentless
* YAML-based Playbooks
* Ansible galaxy

SaltStack

SaltStack is used for data-driven configuration. It is a new approach of infrastructure management built on dynamic communication bus. It is used for data-driven orchestration, remote execution for any infrastructure, and configuration management for any app stack.

Fabric

**Fabric** is a Python-based programming language, which is developed as an API of Python which needs to be imported in Python code in order to configure and manage an infrastructure.