DevOps

Caltech | Center for Technology & Management Education

Post Graduate Program in DevOps

simplilearn

Chef Fundamentals

Learning Objectives

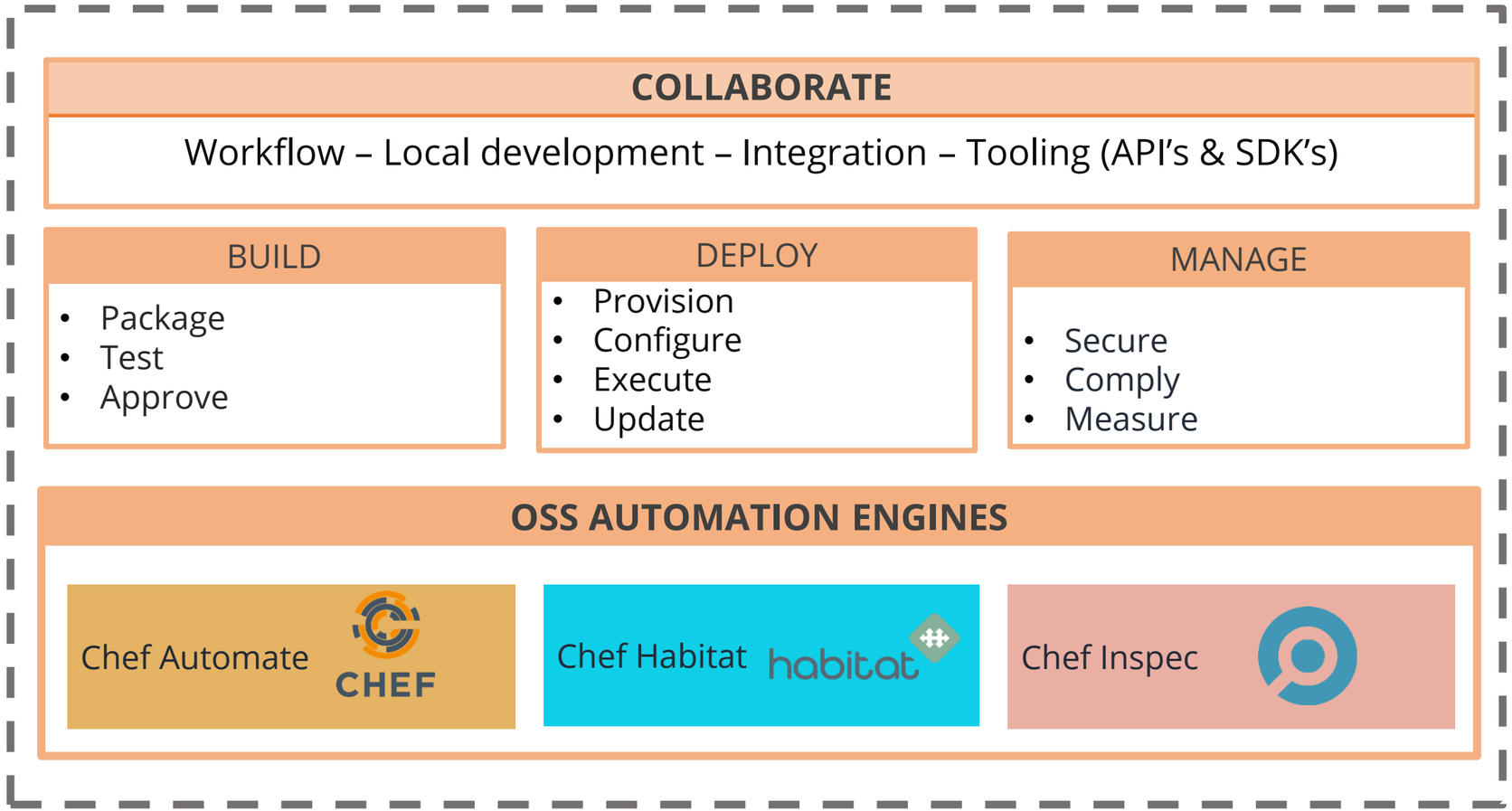
By the end of this lesson, you will be able to:

- Discuss the basic concepts of Chef
- Explain the building blocks of Chef
- Explain Chef architecture in detail
- Install Chef and workstation
- Implement knife and test kitchen setup
- Outline the features of Chef organization and groups

Introduction to Chef

What Is Chef?

Chef is an automation tool that defines infrastructure as code. You can manage both Unix and Windows-based systems using Chef.



Key Metrics of Chef

- Supports multiple platforms like AIX, RHEL/CentOS, FreeBSD, OS X, Solaris, Microsoft Windows and Ubuntu
- Integrates with cloud-based platforms such as Internap, Amazon EC2, Google Cloud Platform, OpenStack, SoftLayer, Microsoft Azure, and Rackspace to automatically provision and configure new machines
- Has an active, smart, and fast growing community support
- Runs both in client-server and solo architecture for managing both infra and software resources

Advantages of Chef

Advantages

1. Accelerates software delivery
2. Improves risk management
3. Increases service resiliency
4. Manages data center and cloud environment
5. Accelerates cloud adoption
6. Delivers all the infrastructure – any app, everywhere, continuously
7. Configure thousands of nodes within minutes using Chef

Building Blocks of Chef

Building Blocks of Chef

1. Recipe
2. Cookbook
3. Resource
4. Attribute
5. File
6. Template
7. Metadata.rb

Recipe

- A collection of attributes or resources that manage the infrastructure and describe a configuration or a policy
- Describes a series of resources that should be in a particular state, that is, packages that should be installed, services that should be running, or files that should be written
- Uses the results of a search query and reads the contents of a data bag

recipe

Cookbook

- A collection of recipes that gets uploaded to the Chef server
- Ensures that recipe gets the desired infrastructure on Chef execution
- Defines a scenario and contains everything that is required to support that scenario:
  - Recipes
  - Attribute values
  - File distributions
  - Templates
  - Extensions to Chef like custom resources and libraries

cookbook

Resource

A resource is a statement of configuration policy that:

- Describes the desired state for a configuration item
- Declares the steps needed to bring that item to the desired state
- Specifies a resource type like package, template, or service
- Lists additional details (also known as resource properties), as necessary
- Is grouped into recipes that describe the working configurations

resources

Attribute

- An attribute is defined in a cookbook (or a recipe) and then used to override the default settings on a node.
- When a cookbook is loaded during a Chef Infra Client run, these attributes are compared to the attributes that are already present on the node.
- Attributes that are defined in attribute files are first loaded according to cookbook order.
- When the cookbook attributes take precedence over the default attributes, Chef Infra Client applies those new settings and values during a Chef Infra Client run on the node.

attributes

File

- A file is a subdirectory within a cookbook that is selected according to the file specificity. This allows different source files to be used based on the hostname, host platform, or platform version.

- Use the **cookbook_file** resource to transfer files from a sub-directory of `COOKBOOK_NAME/files/` to a specified path located on a host that is running Chef Infra Client.

files

Template

- It is a non-static component used to substitute an attribute value into the file to create the final file version.

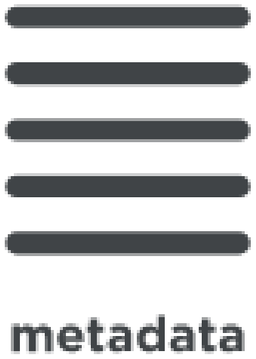
- Templates may contain Ruby expressions and statements, and are a great way to manage configuration files.

templates

Metadata.rb

- It includes the details like name and attributes of the package and allows Chef server to build the run-list.

- The contents of the metadata.rb file provide information that helps Chef Infra Client and Server to deploy cookbooks for each node.



metadata

Chef Architecture and Its Components

Chef Architecture

Chef architecture comprises three main components:

1. Chef Workstation
2. Chef Server
3. Chef Infra Client Nodes

Chef Architecture

Chef Workstation:

- It is the location in the local machine where all the configurations are developed.
- Users can author and test cookbooks using tools such as Test Kitchen and interact with the Chef Infra Server using the knife and Chef command line tools.

Chef Architecture

Chef Server:

- Chef Infra Server stores cookbooks, the policies that are applied to nodes, and metadata that describes each registered node that is being managed by Chef.

- It is a centralized working unit of Chef setup used for uploading configuration files in the system.

Chef Architecture

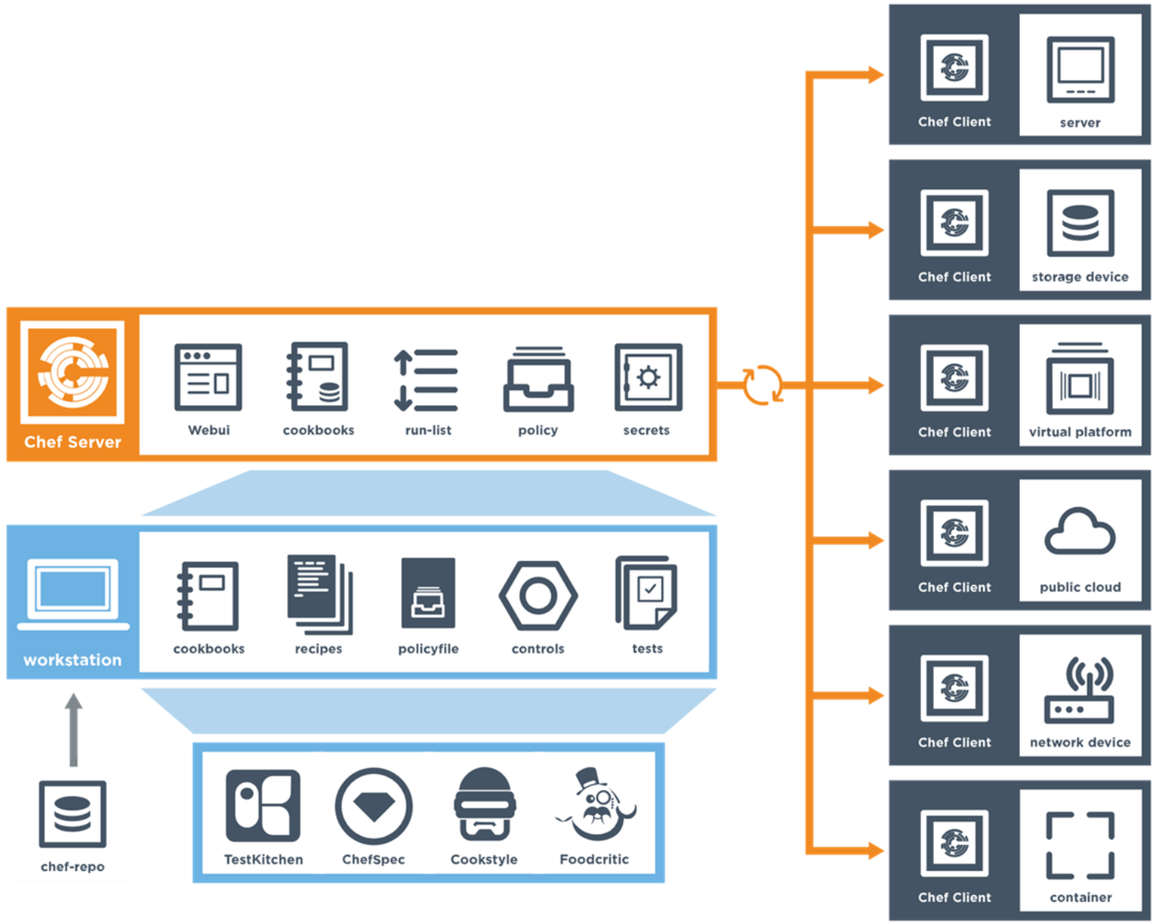
Chef Infra Client Nodes:

- Chef client establishes the communication between Chef server and Chef node.
- The Chef Infra Client is installed on each node and is used to configure the node to its desired state.

Chef Components

Chef Components

The image depicts the relationship between Chef architecture components:



src: chef.io

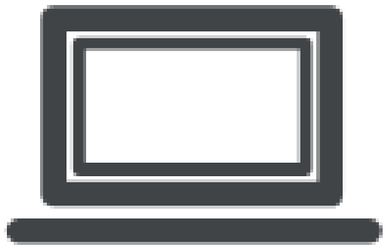
Chef Components

The different components in the Chef Infra are listed below:

1. Workstation
2. Ruby
3. Node
4. Chef Client
5. Chef Server

Workstation

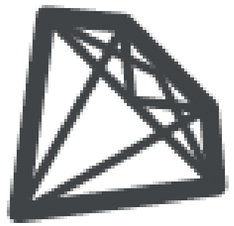
- One or more workstations are configured to allow users to author, test, and maintain cookbooks.
- It runs the Chef Workstation package that includes tools such as, Chef Infra Client, Chef InSpec, Test Kitchen, ChefSpec, Cookstyle.



workstation

Ruby

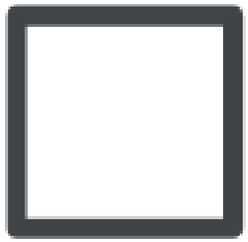
- It is an authoring syntax language for cookbooks.
- It is used to query the cookbooks and recipes in simple patterns.
- Most recipes are simple patterns (blocks that define properties and values that map to specific configuration items like packages, files, services, templates, and users).



Ruby

Node

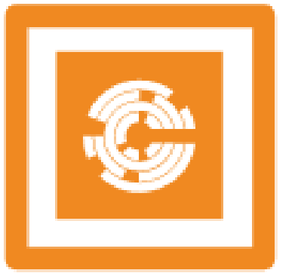
- Any device that is physical, virtual, cloud, or network device
- It is supervisioned under Chef infrastructure to connect server



node

Chef Client

- Configures the node locally by performing the tasks specified in the run-list
- Accesses the required configuration data from the Chef Infra Server during client run
- Uploads updated run data to the Chef Infra Server



Chef Server

- It is the main hub of information
- Cookbooks and policies are uploaded from workstation to the Chef Infra Server
- Client accesses the server from the node point to perform searches on the data run

Chef Supermarket

Chef Supermarket

- A location in which community cookbooks are shared and managed
- Cookbooks in supermarket are accessible by Chef users

Chef Supermarket

Public and Private Supermarket

- The public Chef Supermarket is hosted by Chef software.
- A private Chef Supermarket may be installed on-premise behind the firewall on the internal network.
- Cookbook retrieval from a private Chef Supermarket is often faster than the public Chef Supermarket because of close proximity and fewer cookbooks to resolve.

Chef Supermarket

Installing Private Supermarket

Steps to install private supermarket are given below:

1. Setup Chef Workstation
2. Create a Cookbook
3. Upload a Cookbook
4. Share Cookbook using Knife

Chef Supermarket

Chef Installation

Assisted Practice

Install Chef Server, Client, and Chef Manage

Problem Statement:

You are given a project to set up chef server, chef client, and chef manage on Linux environment.

Assisted Practice: Guidelines

Steps to perform:

1. Download and configure Chef
2. Confirm the installation of Chef Client
3. Install and configure Chef Manage
4. Configure Chef Workstation on your system and confirm the installation

Assisted Practice

Test Kitchen Setup

Problem Statement:

You are given a project to set up your test kitchen to analyze cookbook recipes.

Assisted Practice: Guidelines

Steps to perform:

1. Install test kitchen ruby gem and vagrant gem
2. Create a .kitchen.yml file in the cookbook
3. Setup test kitchen

Assisted Practice
Knife Setup

Problem Statement:

Setup and configure chef knife in the chef repository directory.

Assisted Practice: Guidelines

Steps to perform:

1. Generate chef repository directory
2. Configure Git
3. Configure Knife

Chef Nodes

Chef Nodes

Chef Node can be any device that is physical, virtual, cloud, network device under the Chef Infra management. There are several node types that include:

1. Server
2. Cloud
3. Virtual machine
4. Network device
5. Container

Server

- Server node is an active device capable of sending, receiving, and forwarding information over a communication channel.

- A physical node is any active device attached to a network that can run a Chef Infra Client and allow it to communicate with a Chef Infra Server.

Cloud

- A cloud-based node is hosted in an external cloud-based service, such as Amazon Web Services (AWS), Google Compute Engine, or Microsoft Azure.
- Plugins are available for knife to support external cloud-based services. Knife can use these plugins to create instances on cloud-based services.
- Once created, Chef Infra Client is used to deploy, configure, and maintain those instances.

Virtual Machine

- A virtual machine is a node that runs only as a software implementation.
- It behaves like a physical machine to perform operations.

Network Device

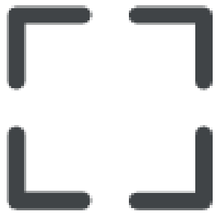
- A network node is any networking devices, such as a switch or router, that is being managed by a Chef Infra Client.

- Chef automates common network configurations, such as physical and logical ethernet link properties and VLANs, on these devices.

Container

- It is an approach to virtualization that allows a single operating system to host many working configurations.

- It is a popular way to manage distributed and scalable applications and services.



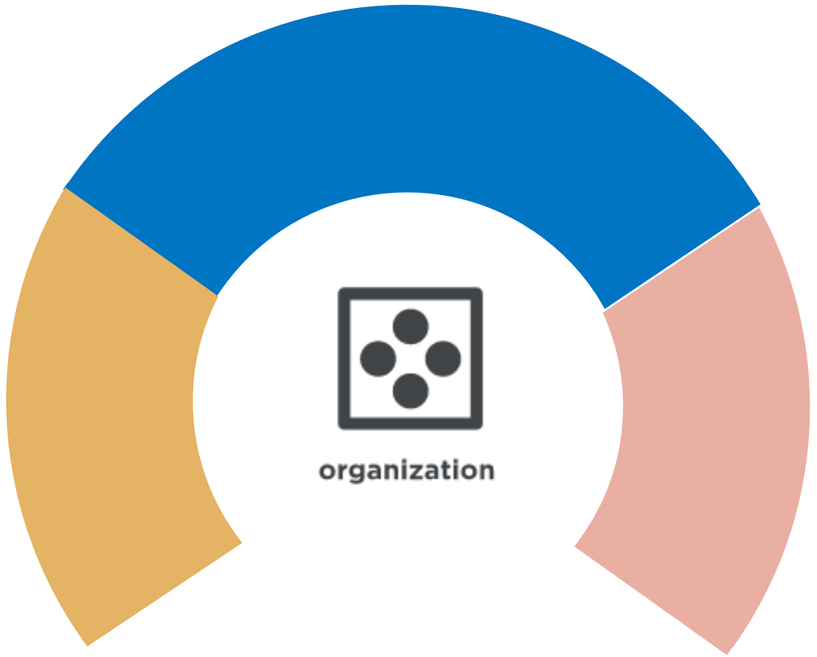
container

Organization, Groups, and Users

Organization

The Chef Infra Server supports multiple organizations and includes a single default organization that is defined during setup.

Organization is a top-level entity for role-based access control in the Chef Infra Server.

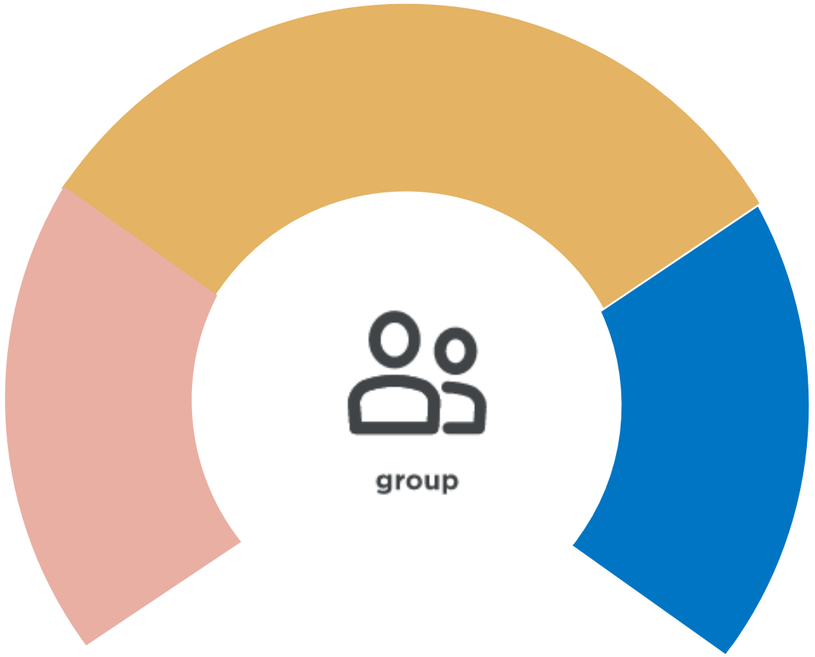


Additional organizations can be created after the initial setup and configuration of the Chef Infra Server.

Group

Determines the tasks available for members of the group who are authorized to perform

Defines access to object types and objects in the Chef Infra Server

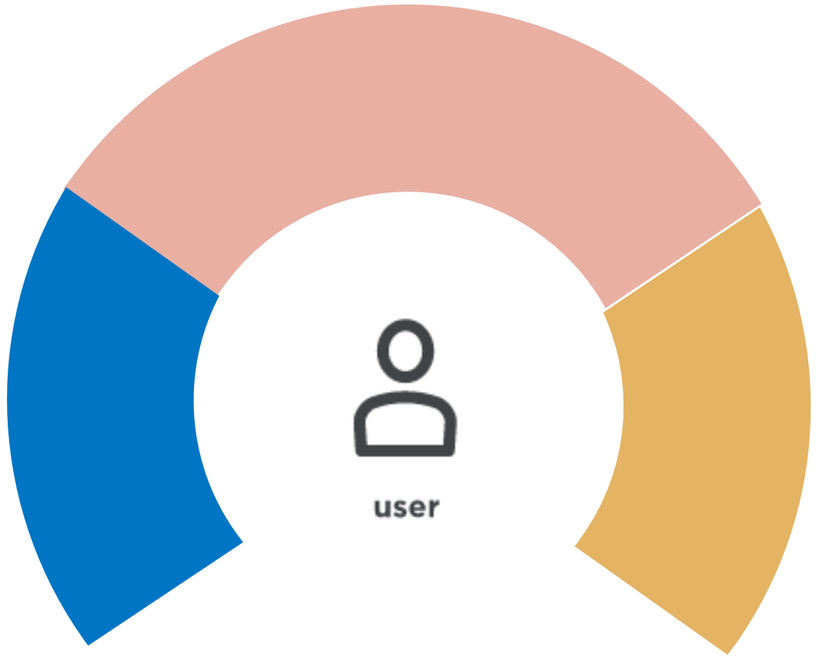


Configures the tasks per-organization

User

Logs the issue from a workstation to Chef management console web user interface

Manages the data uploaded to the Chef Infra Server

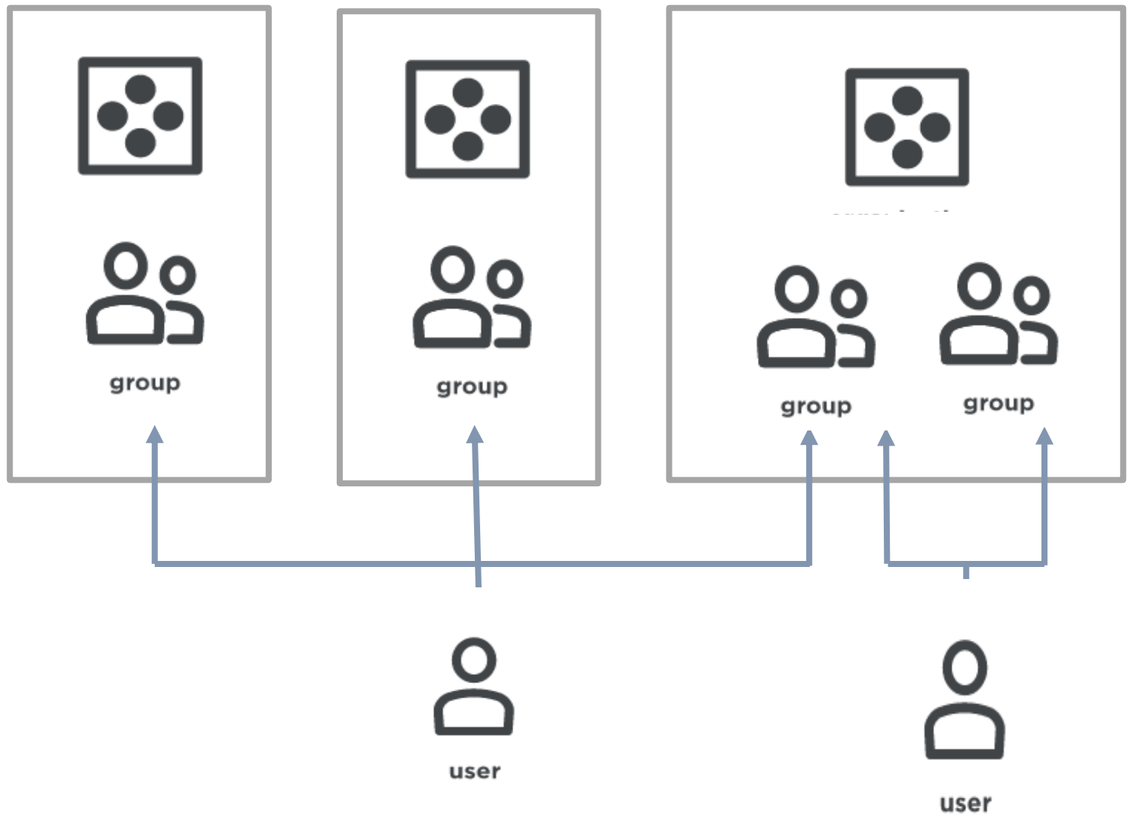


Defines the automatic setup to the admin groups in Chef Infra

Chef Infra Server and Organizations

Chef Infra Server and Organizations

- Each organization has a unique set of groups and users.
- Each organization manages a unique set of nodes, on which a Chef Infra Client is installed and configured.



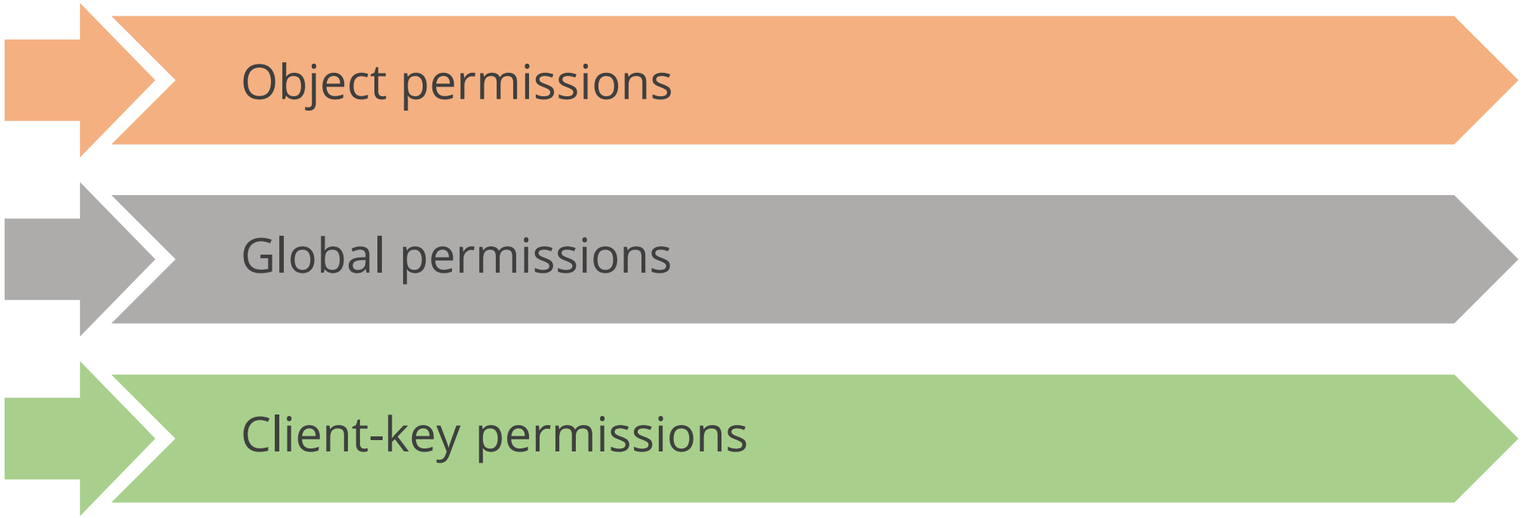
Chef Infra Server and Organizations

A user can belong to multiple organizations under the following conditions:

- Role-based access control is configured per-organization.
- For a single user to interact with the Chef Infra Server using knife from the same Chef-repo, user may need to edit their config.rb file prior to that interaction.

Permissions

The three types of permissions available are:



Object Permissions

Chef Infra Server includes the following object permissions:

1. Delete
2. Grant
3. Read
4. Update

Global Permissions

Chef Infra Server includes the following global permissions:

1. Create
2. List

Client-Key Permissions

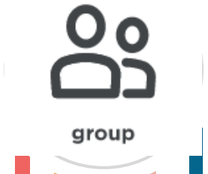
The key should have access to the below Chef-server permissions:

1. Delete
2. Grant
3. Read
4. Update

Chef Infra Server and Groups

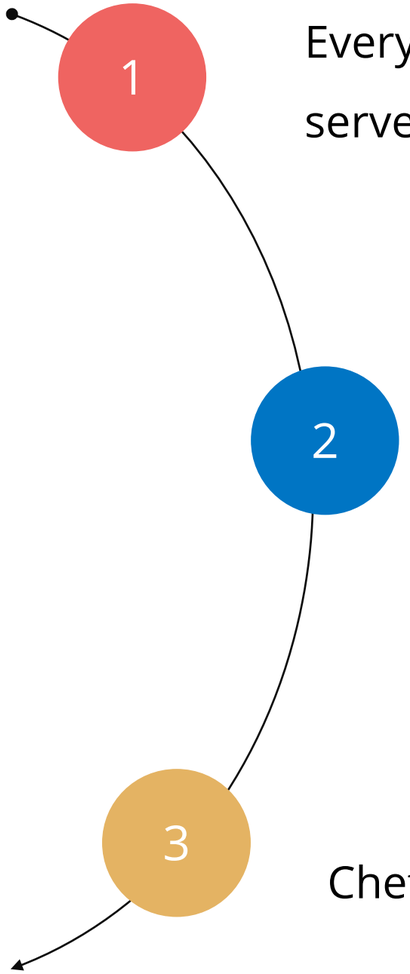
Groups

Chef Infra Server includes the following default groups:



admins	Defines the list of users with admin rights to all the objects
clients	Defines the list of nodes on which a Chef Infra Client is installed and managed
billing_admins	Defines the list of users to manage billing information
Public_read_key_access	Defines the users and clients who have permissions for key-related endpoints
users	Defines the list of users who use knife and the Chef management console to interact with objects and object types

Chef Validators

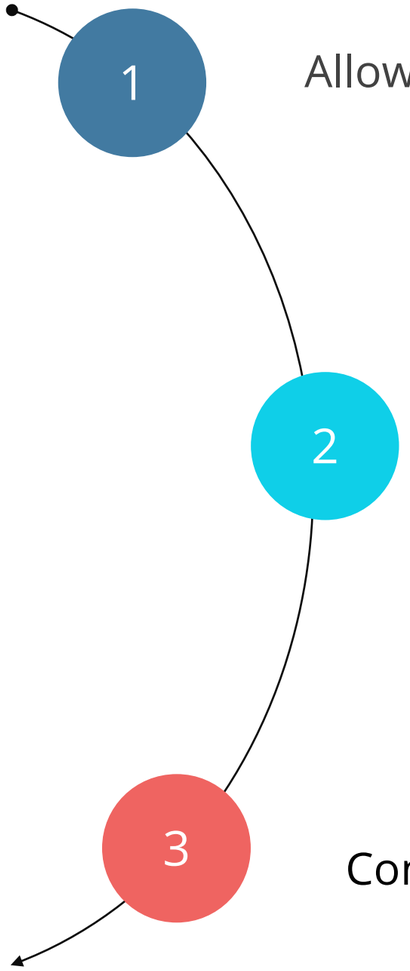


Every request made by the Chef Infra Client to the server must be authenticated using server API and private key.

Chef Infra client authenticates each request using private key located in /etc/Chef/client.pem when client requests to the server.

Chef-validator initiates the server object at the start of a Chef Infra Client run.

Chef Push Job Groups



1. Allows jobs to be run against nodes independent of a Chef Infra Client run
2. Uses the Chef Infra Server API and a Ruby client to initiate all connections to the Chef Infra Server
3. Configures `pushy_job_readers` and `pushy_job_writers` to analyze the job status

Server Admins

Server admins is a global group with the following functionalities:

- Grants permissions to create, read, and update user accounts
- Helps in day-to-day administrations of Chef Infra Server
- Performs user management through knife-user subcommand

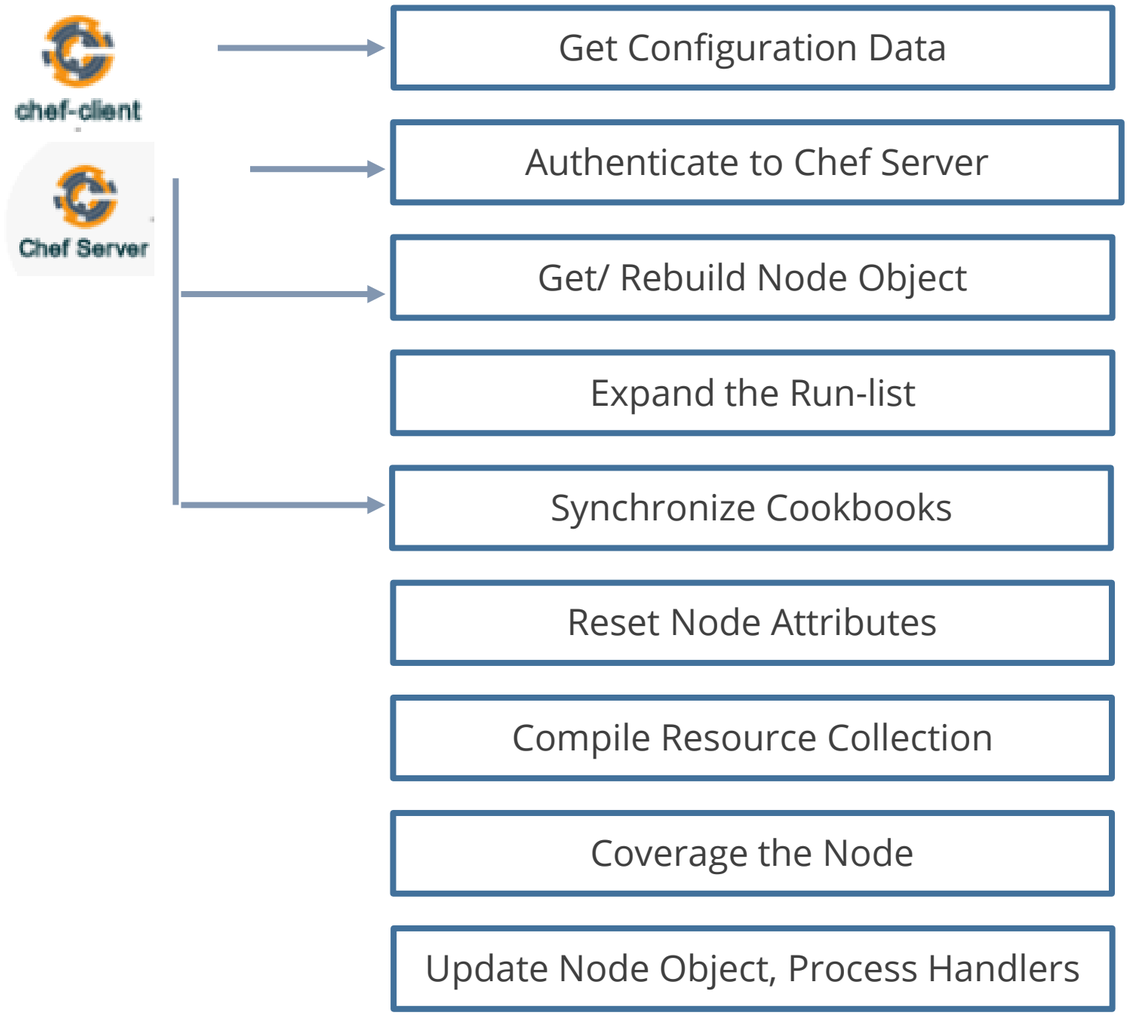
Chef Infra Client Overview

Chef Infra Client

- Chef Infra Client is used to configure the node to its desired state.
- It runs locally on every node that is managed by Chef Infra Server and authenticates the node with Chef Infra Server.
- It also synchronizes cookbooks and builds node objects.

Chef Infra Client Run

The image below describes the steps that Chef Infra Client takes to configure a node during a run.



Chef Infra Client Run

The following sequence of steps occur in every client run:

1. Get configuration data
2. Authenticate to Chef Infra Server
3. Get, rebuild the node object
4. Expand the run-list
5. Synchronize cookbooks

Chef Infra Client Run

The following sequence of steps occur in every client run:

6. Reset node attributes
7. Compile the resource collection
8. Converge the node
9. Update the node object, process exception, and report handlers
10. Stop and wait for the next run

Chef Infra Server Overview

Chef Infra Server

- Chef Infra Server acts as a hub for configuration data and stores cookbooks and metadata on the client.
- It scales the size of the enterprise and distributes configuration throughout the organization.
- Nodes use Chef Infra Client to inform the Chef Infra Server for configuration details, such as recipes, templates, and file distributions

Server Components

The different server components are listed below:

1. Clients
2. Load Balancer
3. Chef Manage
4. Chef Infra Server
5. Bookshelf
6. Message Queues
7. PostgreSQL

Key Takeaways

- Chef runs both on client-server and solo architecture for managing infra and software resources.
- Cookbook is a collection of recipes that gets uploaded on the Chef server.
- Chef Infra Client runs locally on every node managed by Chef Infra Server. It registers and authenticates the node.

Knowledge Check

Knowledge Check

1

What is a run-list in Chef?

A. Outline of roles and recipes in the specific order

B. Recipe in the cookbook

C. Both A and B

D. None of the above

Knowledge Check

1

What is a run-list in Chef?

A. Outline of roles and recipes in the specific order

B. Recipe in the cookbook

C. Both A and B

D. None of the above

The correct answer is **A**

Run-list is an outline of roles and recipes in a specific order in which they have to run. If the run-list has more than one recipe, then the Chef-client will run it only once. It also finds the storage on the Chef server as a part of the node object.

Knowledge Check

2

Which of the following commands uploads a cookbook to the server?

A. knife cookbook upload

B. cookbook upload

C. Both A and B

D. None of the above

Knowledge Check

2

Which of the following commands uploads a cookbook to the server?

A. knife cookbook upload

B. cookbook upload

C. Both A and B

D. None of the above

The correct answer is **A**

One can directly use the command “knife cookbook upload” to upload a cookbook to the Chef server.

Knowledge Check

3

What are data bags in Chef?

A. Global variables in JSON format

B. Local variables in JSON format

C. Both A and B

D. None of the above

Knowledge Check

3

What are data bags in Chef?

A. Global variables in JSON format

B. Local variables in JSON format

C. Both A and B

D. None of the above

The correct answer is **A**

Data bags are global variables stored in the form of JSON data and are accessible from the Chef Server.

Knowledge Check

4

How many handlers are available in Chef?

A. Six

B. Four

C. Three

D. None of the above

Knowledge Check

4

How many handlers are available in Chef?

A. Six

B. Four

C. Three

D. None of the above

The correct answer is **C**

There are three types of handlers in Chef that include, exception handler, start handler, and report handler.

Knowledge Check

5

Which of the following is a component of Chef?

A. Workstation

B. Chef Node

C. Chef Client

D. All the above

Knowledge Check

5

Which of the following is a component of Chef?

A. Workstation

B. Chef Node

C. Chef Client

D. All the above

The correct answer is **D**

Chef comprises Workstation, Chef Node, Chef Client, Chef Server, and Chef Supermarket.