
DEPLOYING COOKBOOKS

The Deploying Cookbooks badge is awarded when someone proves that they understand how to use Chef server to manage nodes and ensure they're in their expected state.

Candidates must show:

- An understanding of the chef-client run phases.
- An understanding of using Roles and Environments.
- An understanding of maintaining cookbooks on the Chef server.
- An understanding of using knife.
- An understanding of using how bootstrapping works.
- An understanding of what policy files are.
- An understanding of what Chef solo is.
- An understanding of using search.
- An understanding of databags.

Here is a detailed breakdown of each area.

ANATOMY OF A CHEF RUN

CHEF-CLIENT OPERATION

Candidates should understand:
chef-client modes of operation
Configuring chef-client

COMPILE VS EXECUTE

Candidates should understand:
What happens during the 'compile' phase?
What happens during the 'execute' phase?
What happens when you place some Ruby at the start of a recipe?
What happens when you place some Ruby at the end of a recipe?
When are attributes evaluated?
What happens during a 'node.save' operation?"

RUN_STATUS

Candidates should understand:
How can you tap into the chef-client run?
What is 'run_status'?
What is 'run_state'?
What is the 'resource collection'?

AUTHENTICATION

Candidates should understand:

How does the chef-client authenticate with the Chef Server?
Authentication and using NTP

CHEF COMPILE PHASE

Candidates should understand:

Do all cookbooks always get downloaded?

What about dependencies, and their dependencies?

What order do the following get loaded - libraries, attributes, resources/providers, definitions, recipes?

CONVERGENCE

Candidates should understand:

What happens during the execute phase of the chef-client run?

The test/repair model

When do notifications get invoked?

What happens if there are multiple start notifications for a particular resource?

When is node data sent to the Chef Server?

WHY-RUN

Candidates should understand:

What is the purpose of 'why-run'?

How do you invoke a 'why-run'?

What are limitations of doing a why run?

ENVIRONMENTS

WHAT IS AN ENVIRONMENT/USE CASES

Candidates should understand:

What is the purpose of an Environments?

What is the '_default' environment?

What information can be specified in an Environment?

What happens if you do not specify an Environment?

Creating environments in Ruby

Creating environments in JSON

Using environments within a search

ATTRIBUTE PRECEDENCE AND COOKBOOK CONSTRAINTS

Candidates should understand:

What attribute precedence levels are available for Environments

Overriding Role attributes

Syntax for setting cookbook constraints.

How would you allow only patch updates to a cookbook within an environment?

SETTING AND VIEWING ENVIRONMENTS

Candidates should understand:

How can you list Environments?

How can you move a node to a specific Environment?

Using ``knife exec`` to bulk change Environments.

Using `'chef_environment'` global variable in recipes

Environment specific knife plugins, e.g. ``knife flip``

Bootstrapping a node into a particular Environment

ROLES

USING ROLES

Candidates should understand:

What is the purpose of a Role?

Creating Roles

Role Ruby & JSON DSL formats

Pros and Cons of Roles

The Role cookbook pattern

Creating Role cookbooks

Using Roles within a search

SETTING ATTRIBUTES AND ATTRIBUTE PRECEDENCE

Candidates should understand:

What attribute precedence levels are available for Roles

Setting attribute precedence

BASE ROLE & NESTED ROLES

Candidates should understand:

What are nested roles?

Whats the purpose of a 'base' role?

USING KNIFE

Candidates should understand:

The ``knife role`` command

How would you set the `run_list` for a node using ``knife``?

Listing nodes

View role details

Using Roles within a search

UPLOADING COOKBOOKS TO CHEF SERVER

USING BERKSHELF

Candidates should understand:

The Berksfile & Berksfile.lock

``berks install`` and ``berks upload``

Where are dependant cookbooks stored locally?

Limitations of using knife to upload cookbooks

Listing cookbooks on Chef Server using knife

What happens if you try to upload the same version of a cookbook?

How can you upload the same version of a cookbook?

Downloading cookbooks from Chef Server

Bulk uploading cookbooks using knife

USING KNIFE

BASIC KNIFE USAGE

Candidates should understand:

How does knife know what Chef Server to interact with?

How does knife authenticate with Chef Server

How/When would you use ``knife ssh`` & ``knife winrm``?

Verifying ssl certificate authenticity using knife

Where can/should you run the ``knife`` command from?

KNIFE CONFIGURATION

Candidates should understand:

How/where do you configure knife?

Common options - `cookbook_path`, `validation_key`, `chef_server_url`, `validation_key`

Setting the chef-client version to be installed during bootstrap

Setting defaults for command line options in knife's configuration file

Using environment variables and sharing knife configuration file with your team

Managing proxies

KNIFE PLUGINS

Candidates should understand:

Common knife plugins-

What is 'knife ec2' plugin

What is 'knife windows' plugin

What is 'knife block' plugin?

What is 'knife spork' plugin?

Installing knife plugins

TROUBLESHOOTING

Candidates should understand:

Troubleshooting Authentication

Using `knife ssl check` command

Using `knife ssl fetch` command

Using '-VV' flag

Setting log levels and log locations

BOOTSTRAPPING

USING KNIFE

Candidates should understand:

Common 'knife bootstrap' options - UserName, Password, RunList, and Environment

Using `winrm` & `ssh`

Using knife plugins for bootstrap - `knife ec2 ..`, `knife bootstrap windows ...`

BOOTSTRAP OPTIONS

Candidates should understand:

Validator' vs 'Validatorless' Bootstraps

Bootstrapping in FIPS mode,

What are Custom Templates

UNATTENDED INSTALLS

Candidates should understand:

Configuring Unattended Installs

What conditions must exist for unattended install to take place?

FIRST CHEF-CLIENT RUN

Candidates should understand:

How does authentication work during the first chef-client run?

What is 'ORGANIZATION-validator.pem' file and when is it used?

What is the 'first-boot.json' file?

POLICY FILES

BASIC KNOWLEDGE AND USAGE

Candidates should understand:

What are policy files, and what problems do they solve?

Policy file use cases?

What can/not be configured in a policy file?

Policy files and Chef Workflow

SEARCH

BASIC SEARCH USAGE

Candidates should understand:

What information is indexed and available for search?

Search operators and query syntax

Wildcards, range and fuzzy matching

SEARCH USING KNIFE AND IN A RECIPE

Candidates should understand:

Knife command line search syntax

Recipe search syntax

FILTERING RESULTS

Candidates should understand:

How do you filter on Chef Server

Selecting attributes to be returned

CHEF SOLO

WHAT CHEF SOLO IS

Candidates should understand:

Advantages & disadvantages of Chef-solo vs Chef Server

Chef-solo executable and options

Cookbooks, nodes and attributes

Using Data Bags, Roles & Environments

Chef-solo run intervals

Retrieving cookbooks from remote locations

Chef-solo and node object

DATA BAGS

WHAT IS A DATA_BAG

Candidates should understand:

When might you use a data_bag?

Indexing data_bags

What is a data_bag?

Data_bag and Chef-solo

DATA_BAG ENCRYPTION

Candidates should understand:

How do you encrypt a data_bag
What is Chef Vault

USING DATA_BAGS

Candidates should understand:
How do you create a data_bag?
How can you edit a data_bag