



Chef Fundamentals

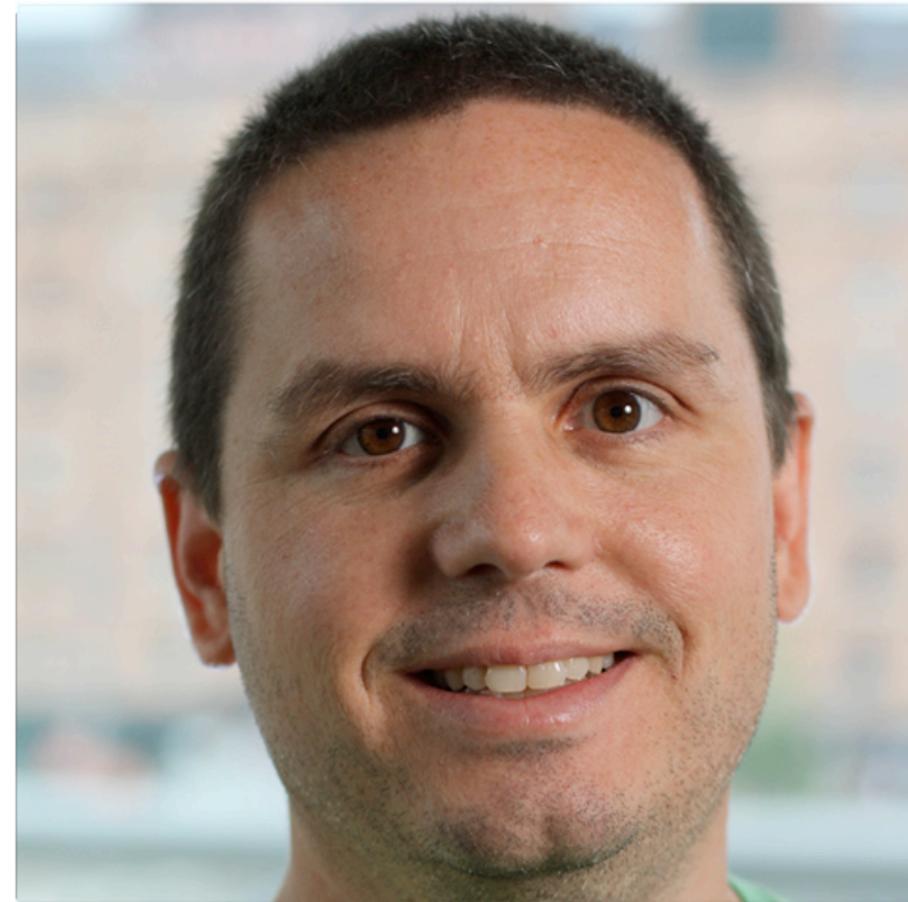
Chef Client Run and Expanding Our Cookbook

Chef Fundamentals Webinar Series - Module 4

training@opscode.com

Nathen Harvey

- Technical Community Manager at Opscode
- Co-host of the Food Fight Show Podcast
- @nathenharvey
- nharvey@opscode.com



Quick Recap

Checkpoint

- In the last module we
 - bootstrapped a node using `knife bootstrap`
 - wrote a simple cookbook to deploy a simple webpage

Where'd my Node go?

- You still need an Ubuntu 10.04+ machine to manage
 - Launch a new instance in the training lab
 - Fire up a new Vagrant
 - Launch an new AMI Instance

But it might not have worked

- In some instances, the apt-cache was out of date
- You can fix this!
 - `sudo apt-get update`

Fix it the Chef way!

- Create an apt cookbook
- Use an execute resource
- Add to the run list

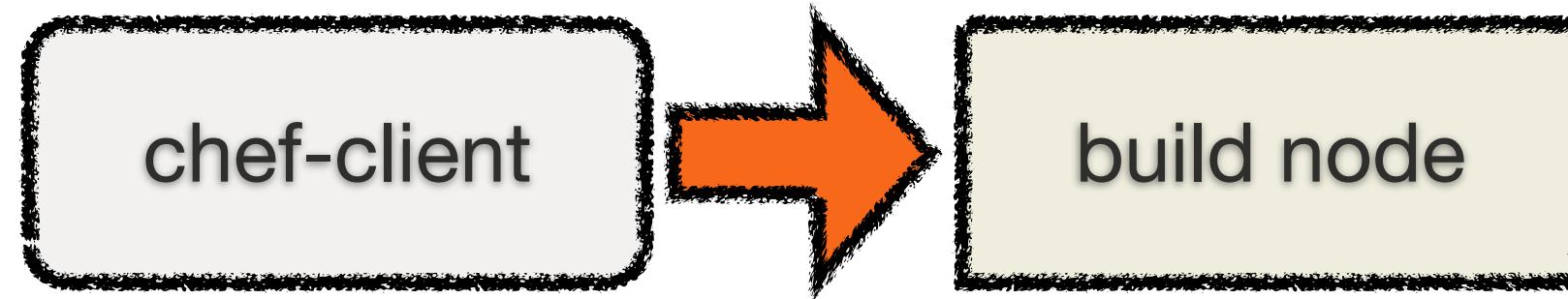
Dissecting your first chef-client run

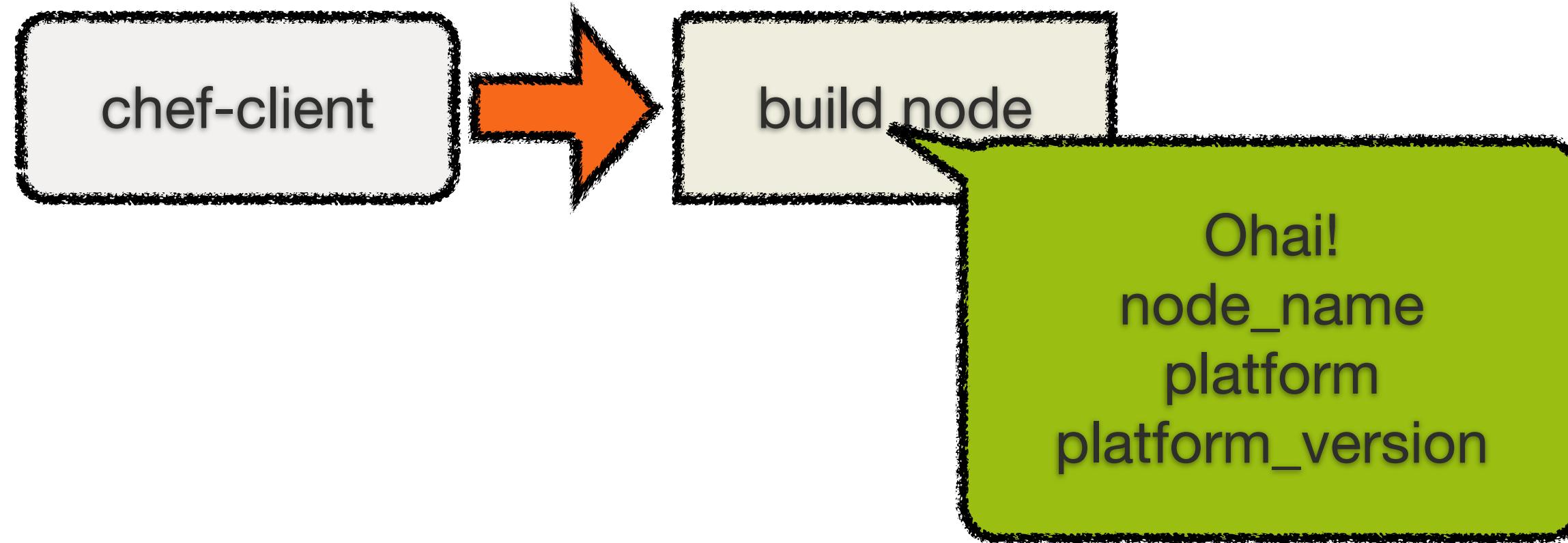
The Anatomy of a Chef run

Objectives

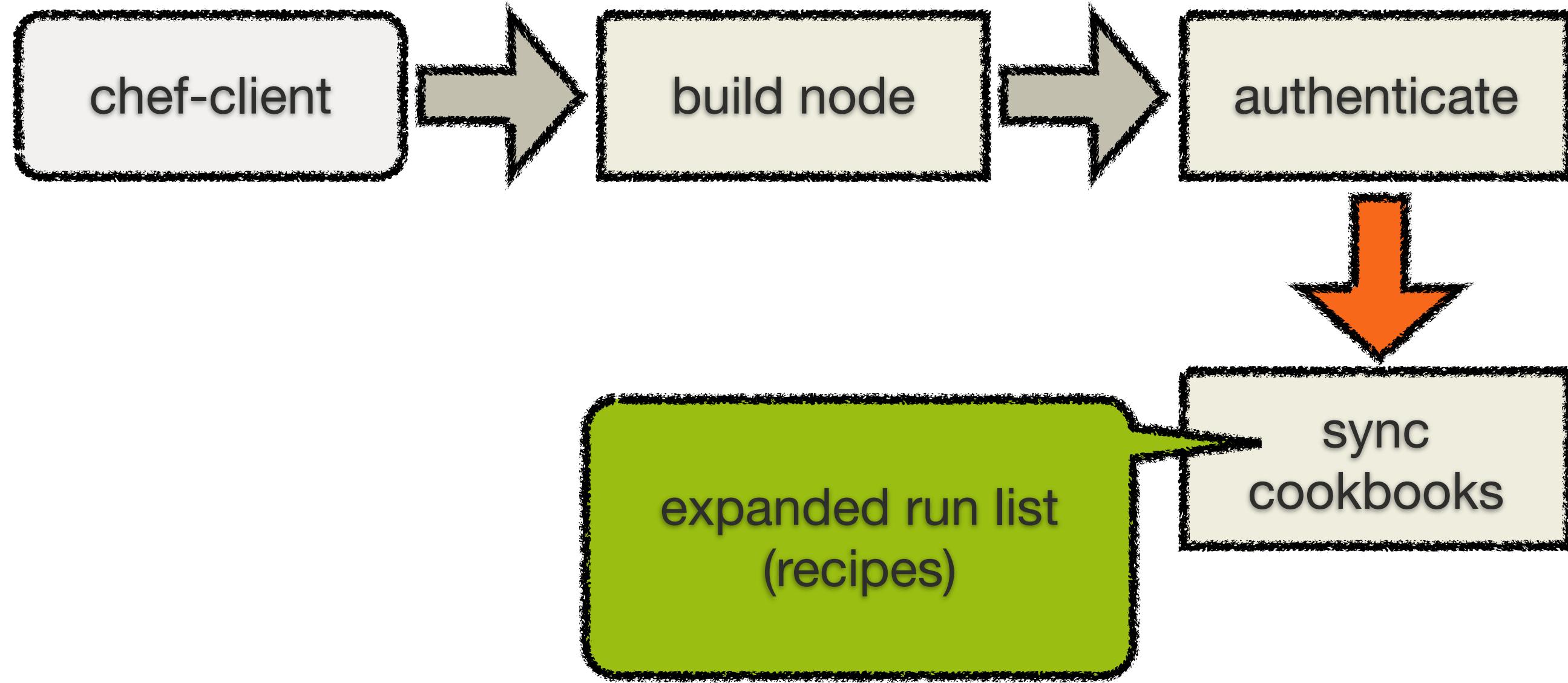
- Describe the steps of a chef-client run
- Describe the basic security model of Chef

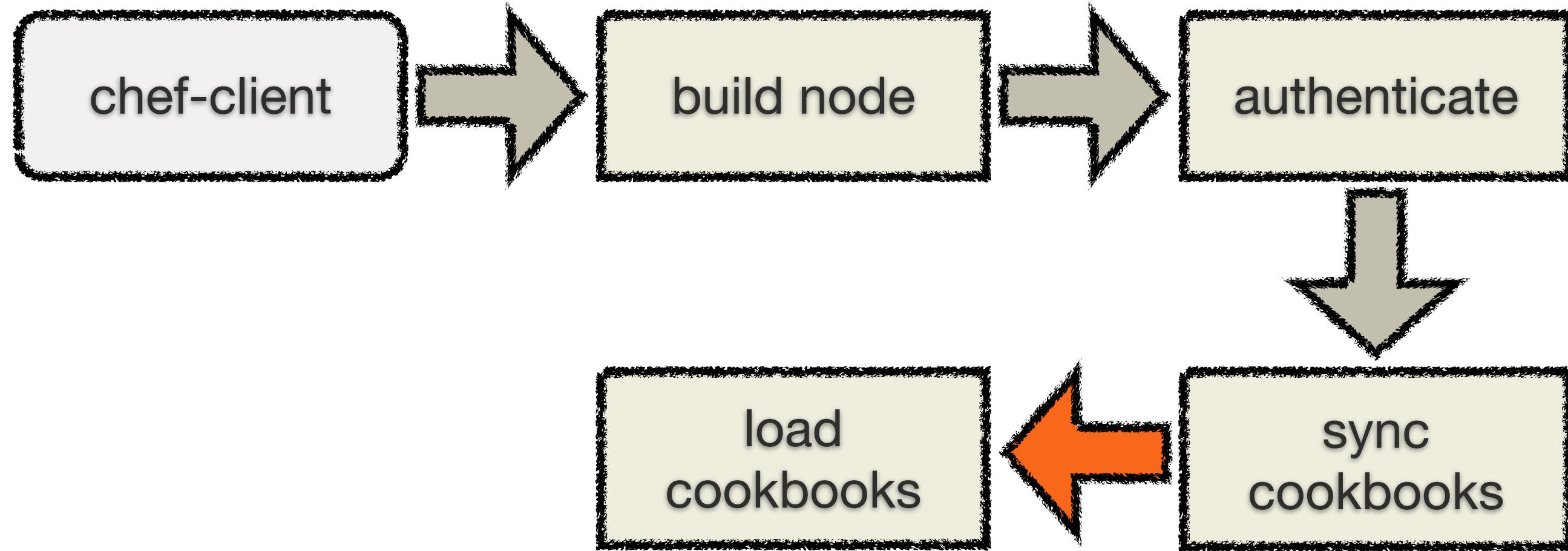
chef-client

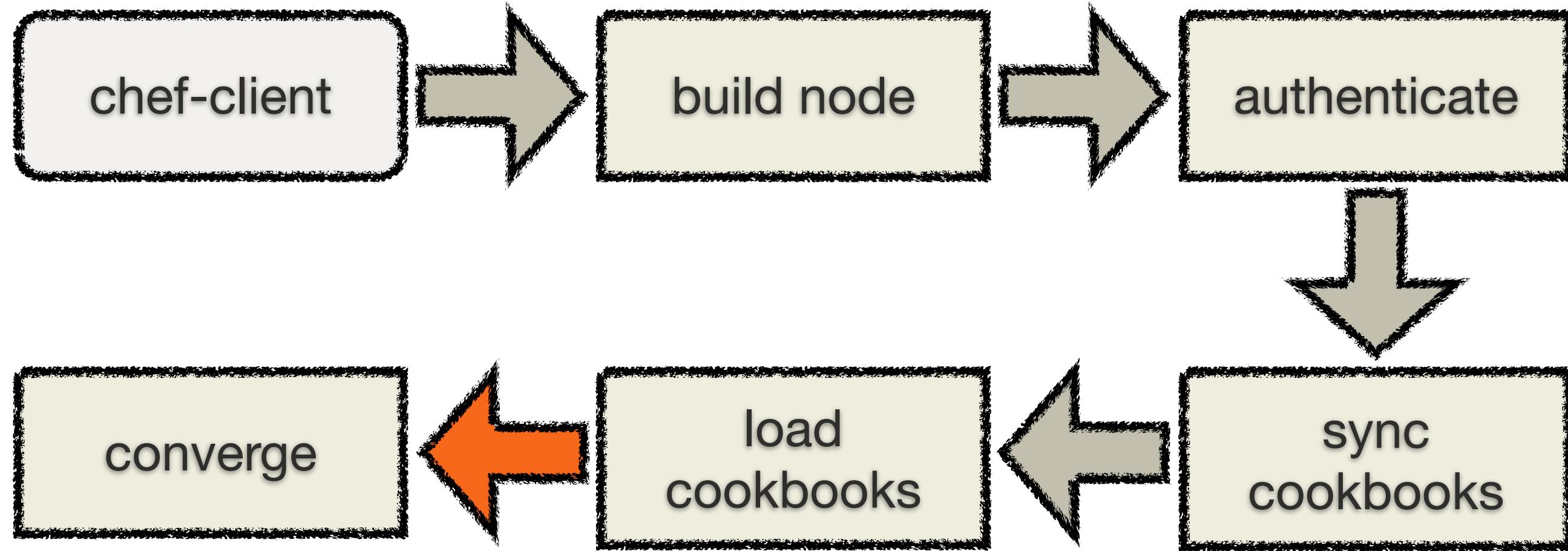


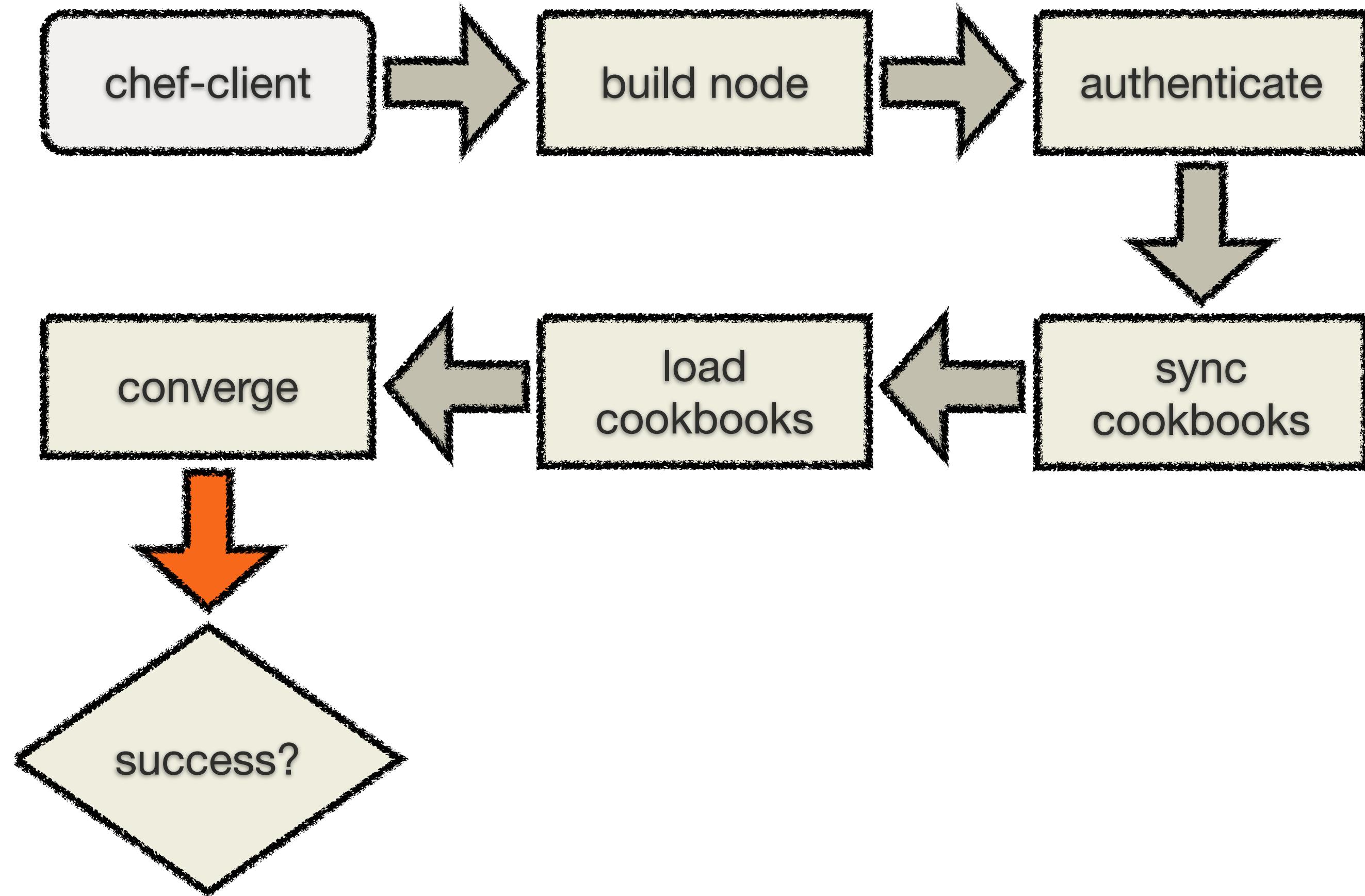


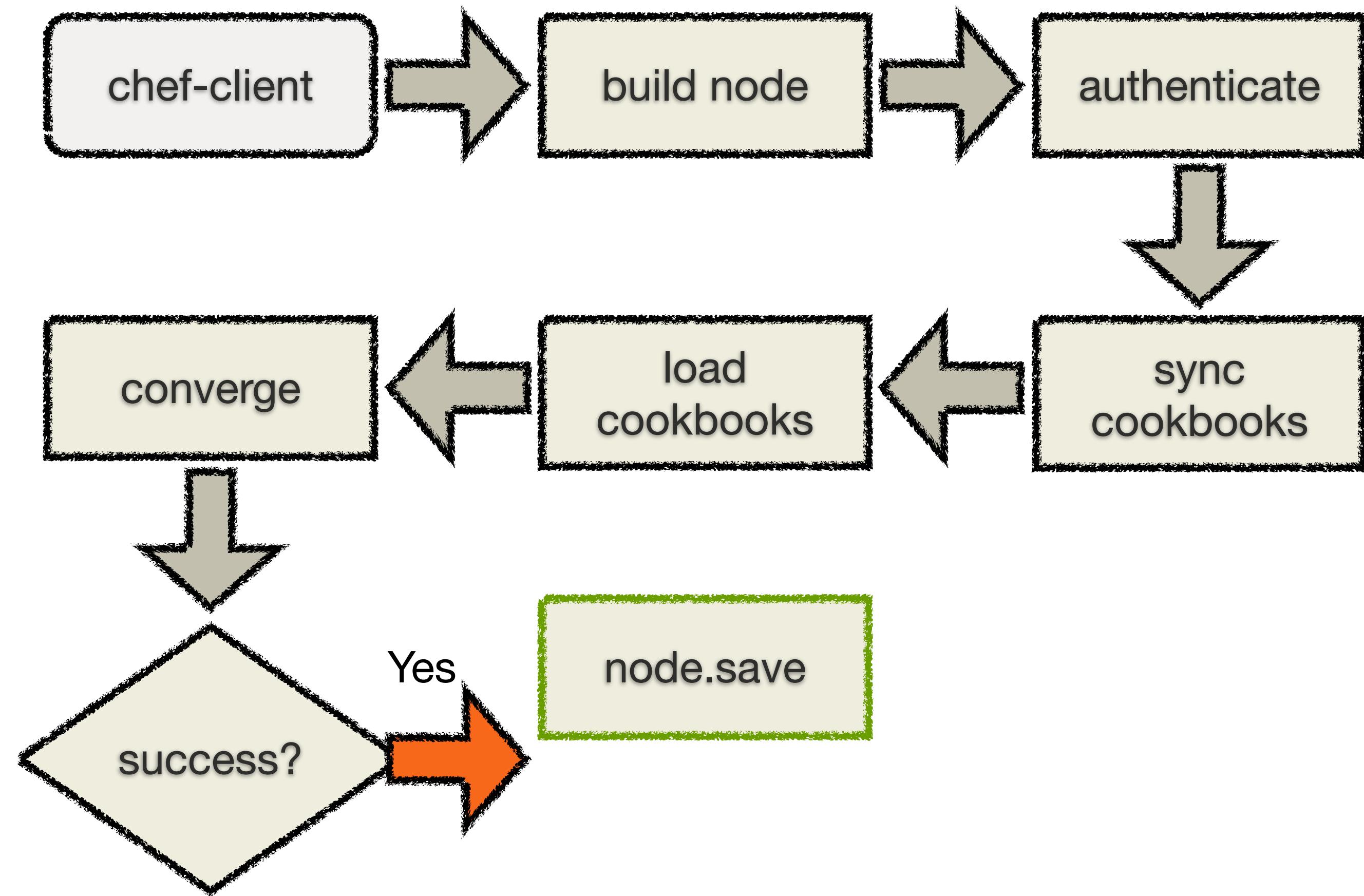


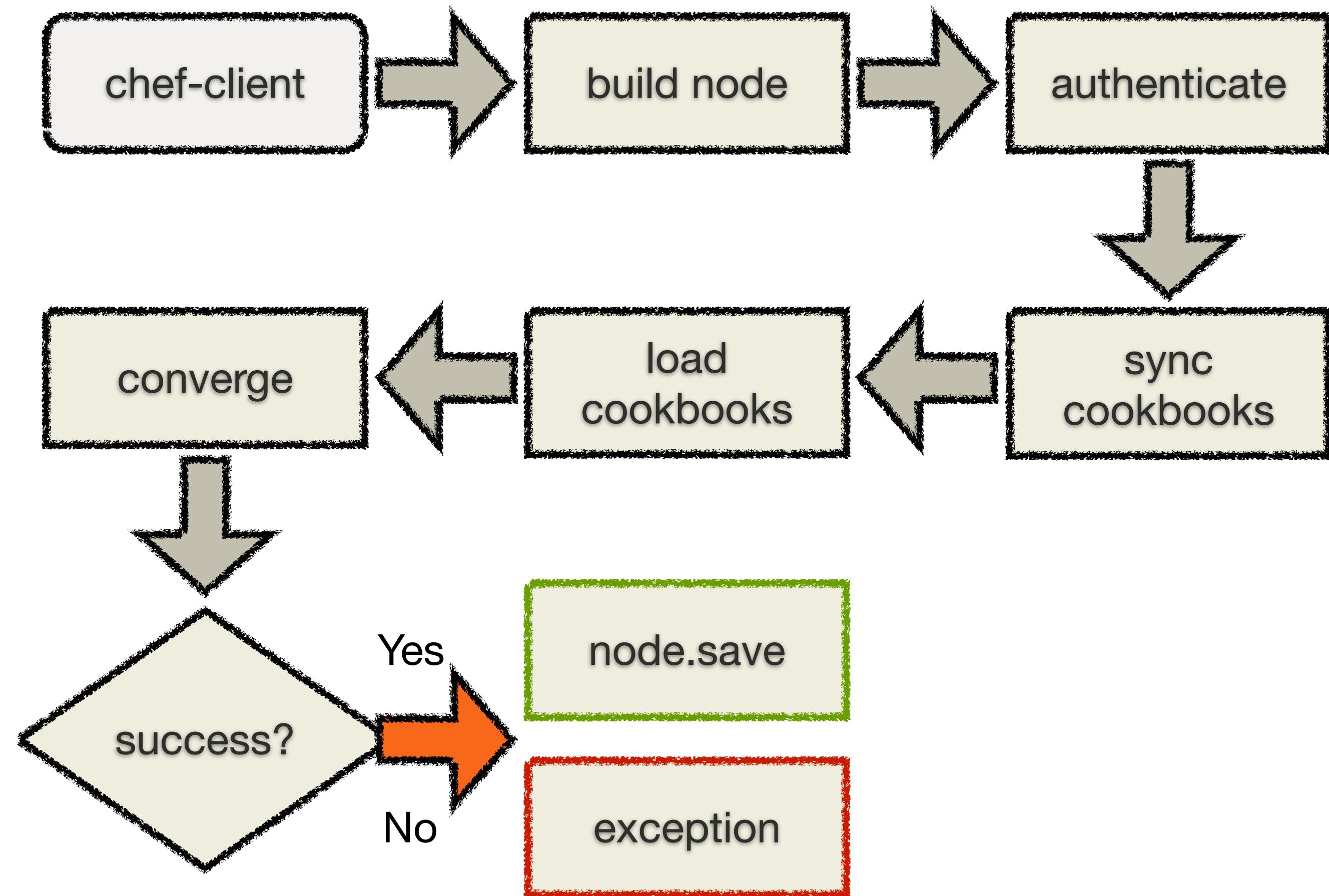


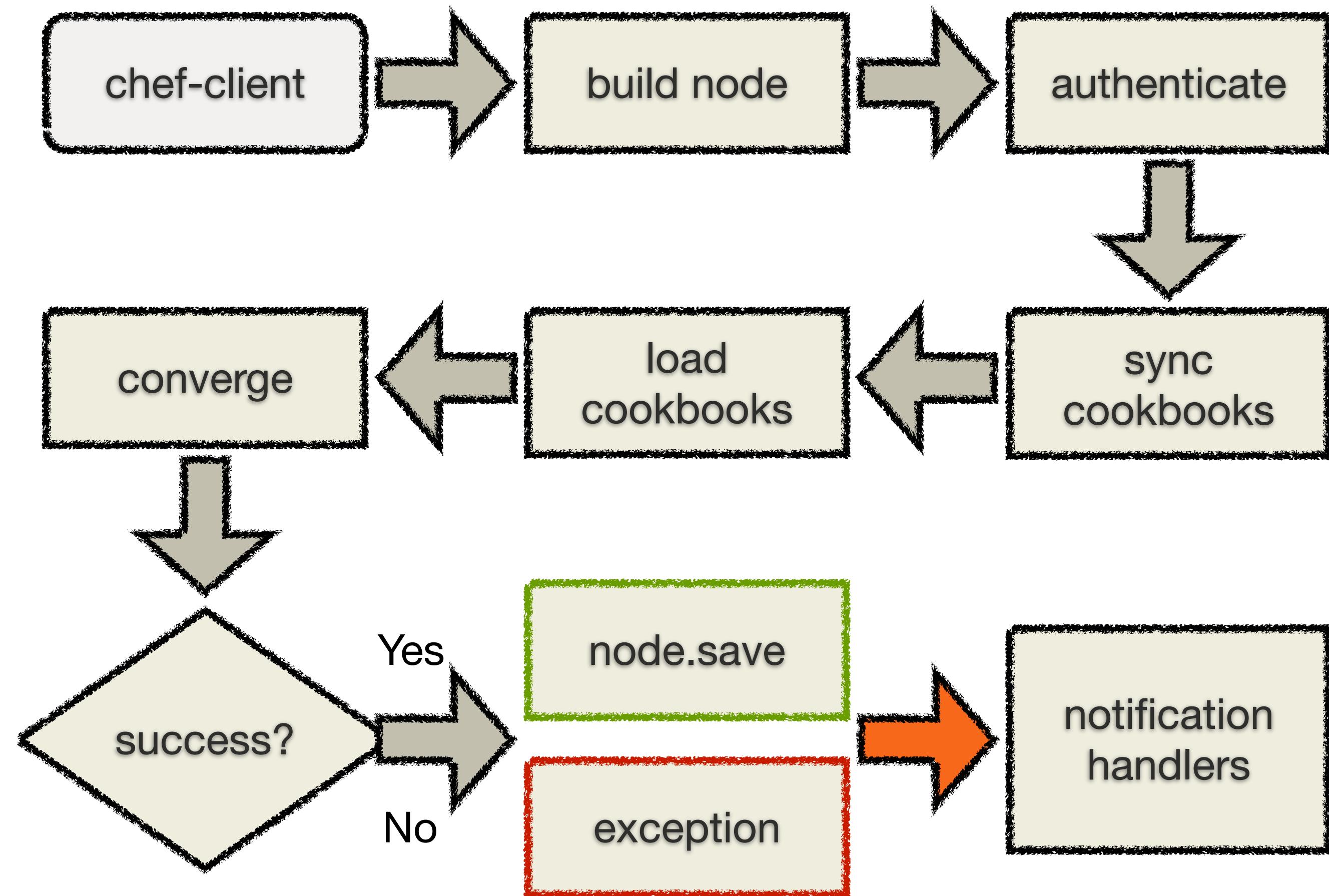






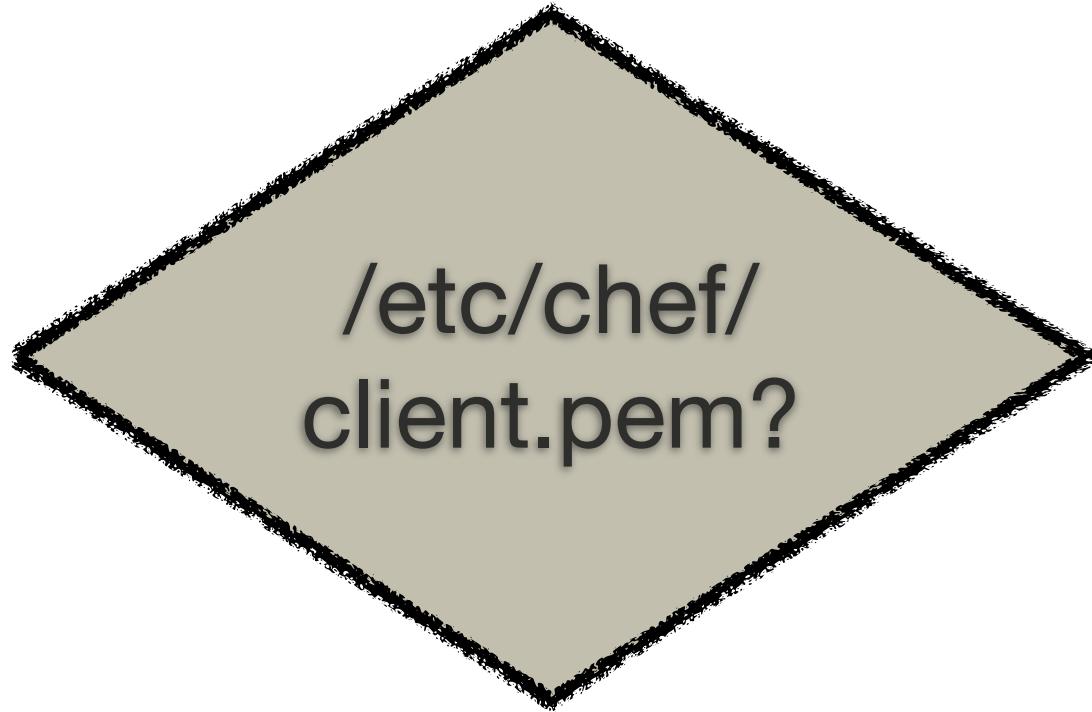




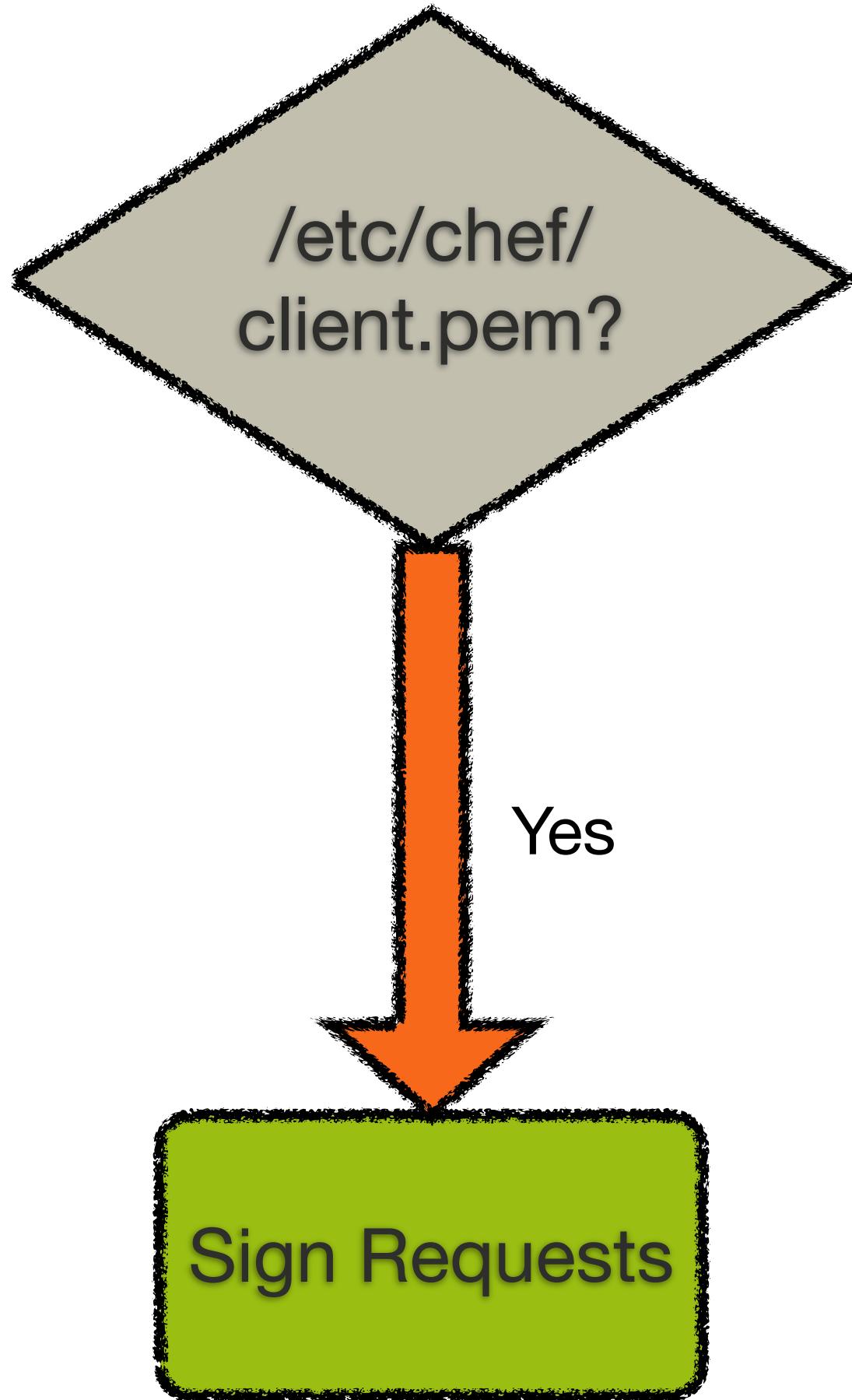


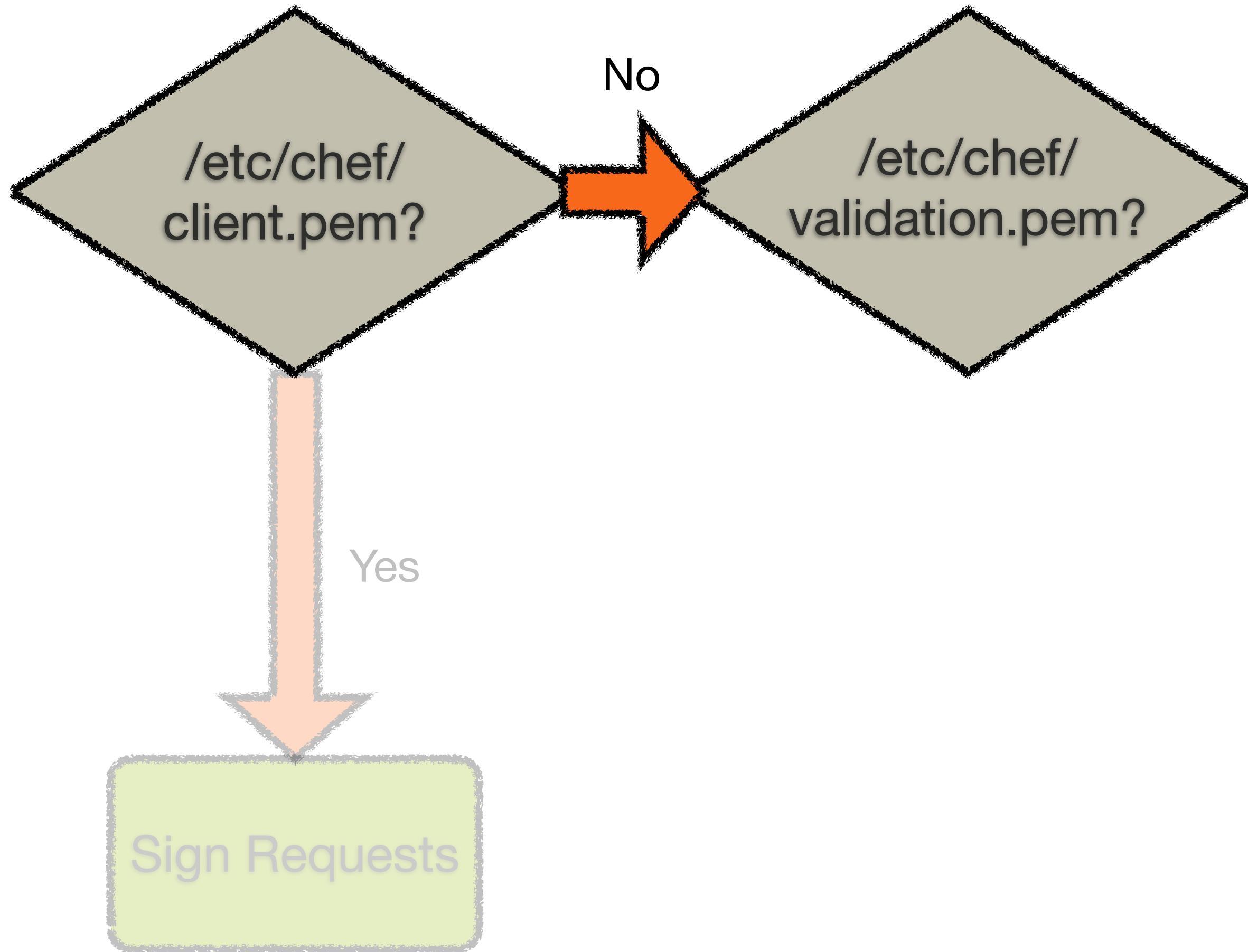
Private Keys

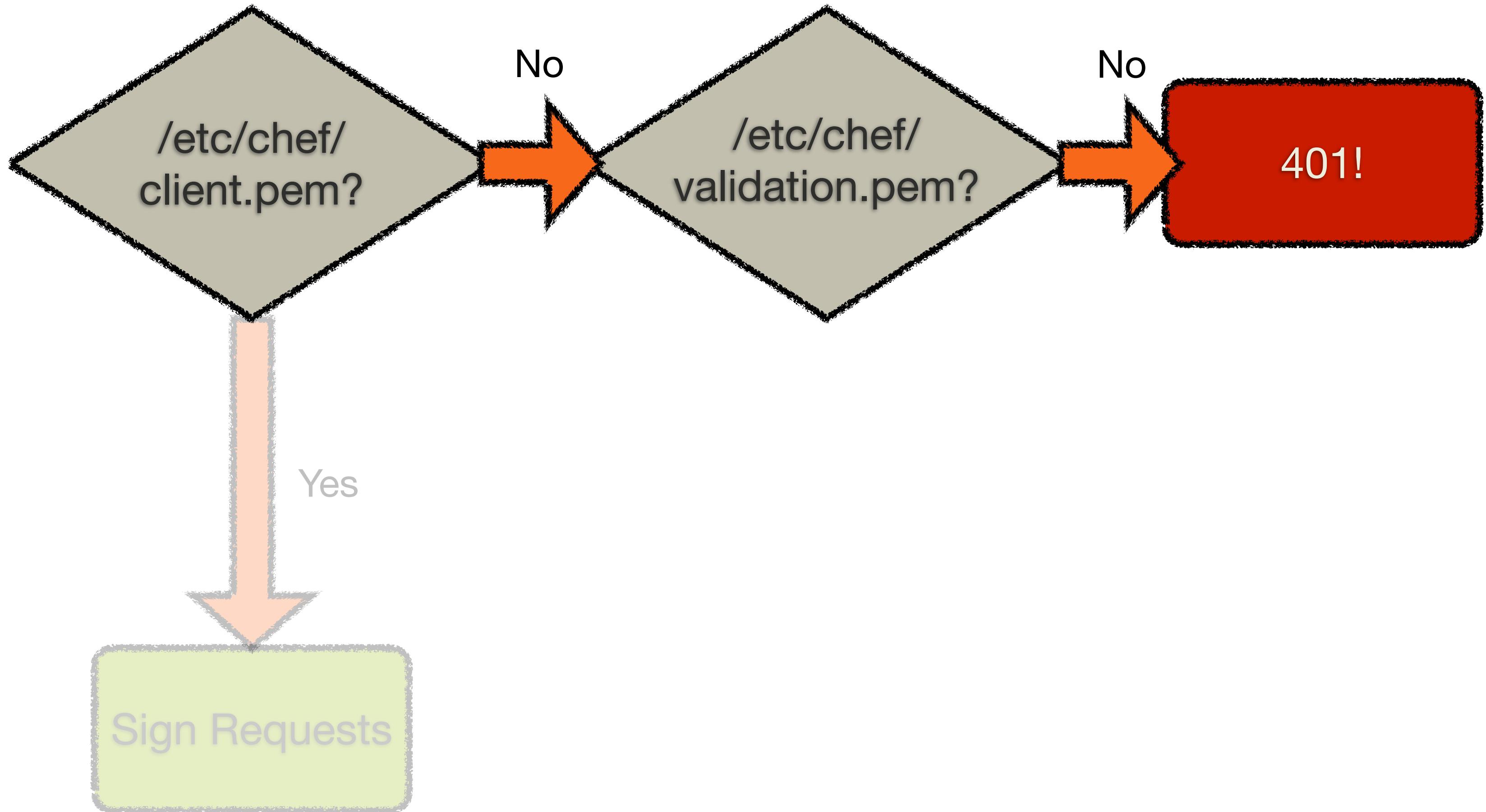
- Chef Server requires keys to authenticate.
 - client.pem - private key for API client
 - validation.pem - private key for ORGNAME-validator
- Next, let's see how those are used...

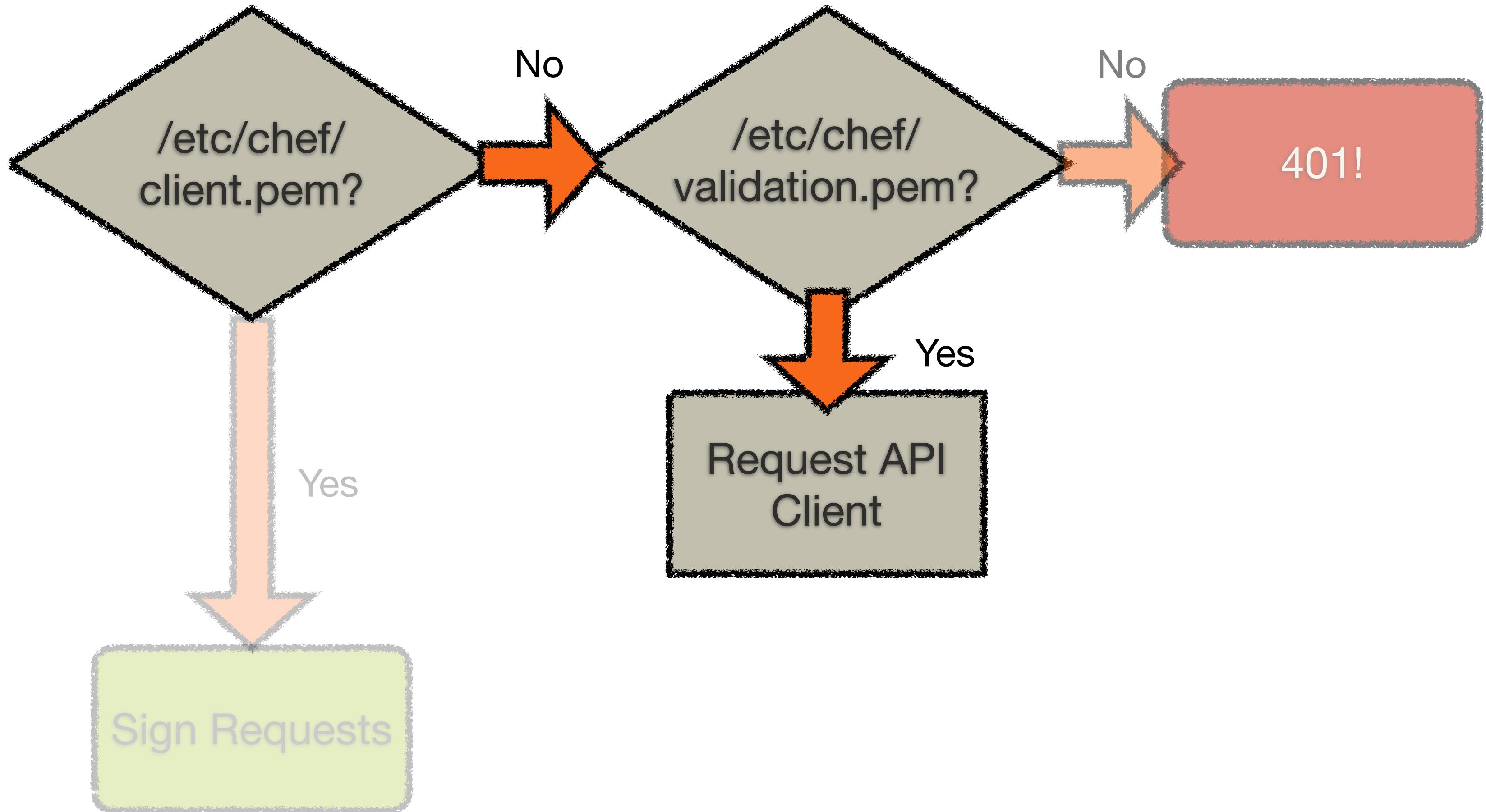


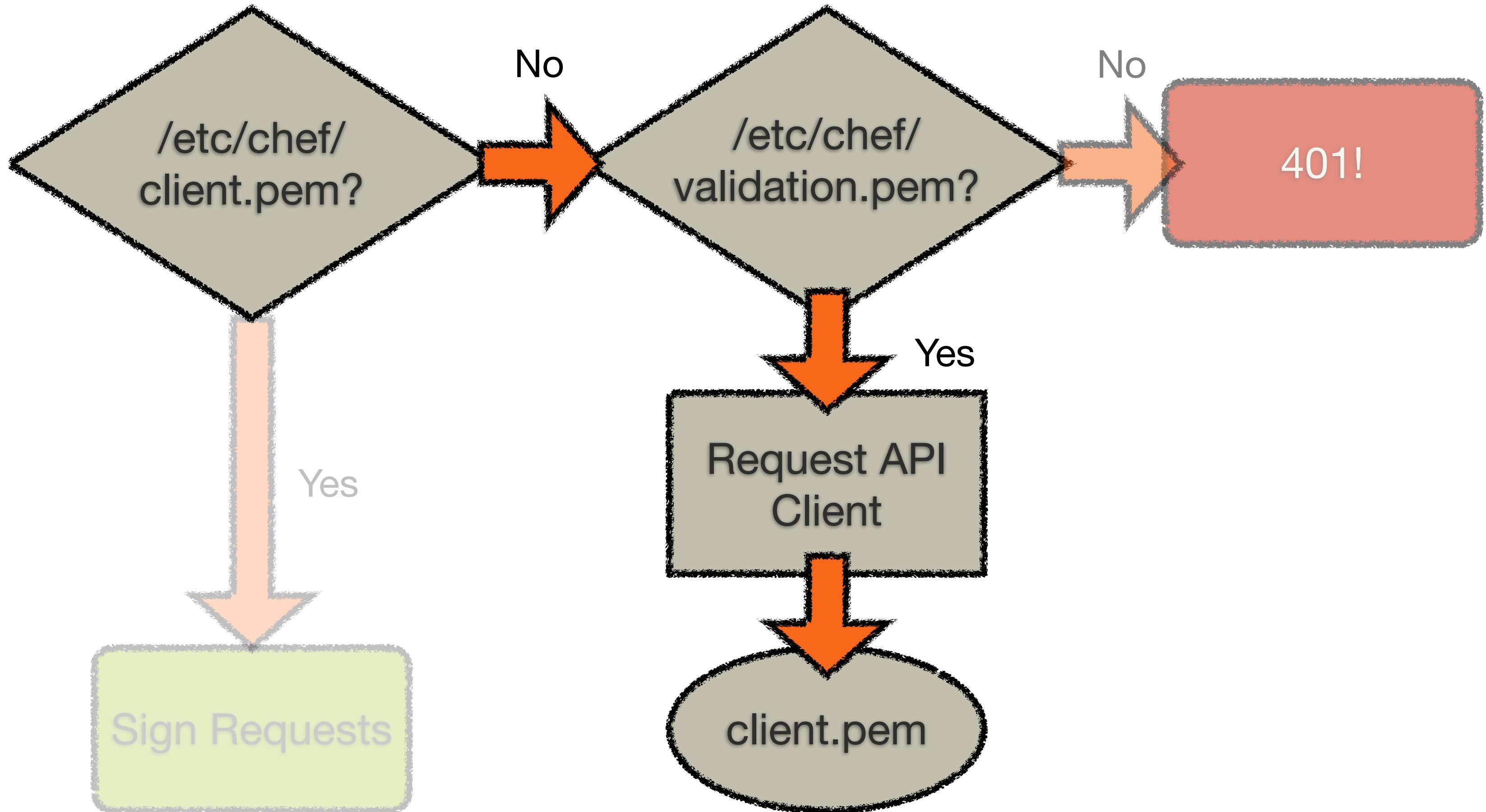
/etc/chef/
client.pem?

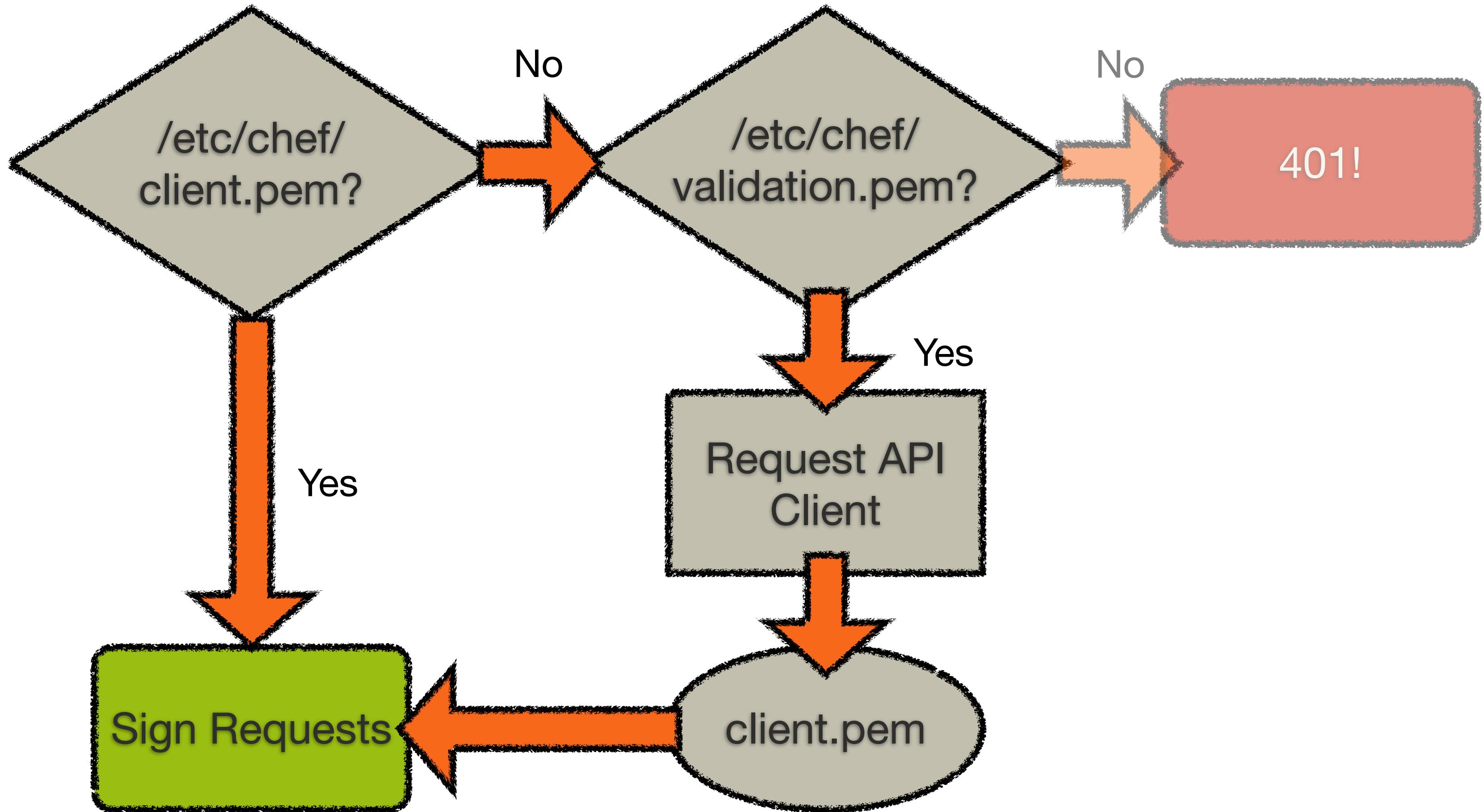












Compile and Execute

- Compile a Resource Collection
- Execute the Resources in that Collection

Introducing the Node object

Attributes & Search

Lesson Objectives

- After completing the lesson, you will be able to
 - Explain what the Node object represents in Chef
 - List the Nodes in an organization
 - Show details about a Node
 - Describe what Node Attributes are
 - Retrieve a node attribute directly, and via search

What is the Node object

- A node is any physical, virtual, or cloud machines that is configured to be maintained by a Chef
- When you are writing Recipes, the Node object is always available to you.

Exercise: List nodes

```
$ knife node list
```

target1

Exercise: List clients

```
$ knife client list
```

ORGNAME-validator
target1

Each node must have a unique name

- Every node must have a unique name within an organization
- Chef defaults to the *Fully Qualified Domain Name* of the server, i.e. in the format `server.domain.com`
- We overrode it to "target1" to make typing easier

Exercise: Show node details

```
$ knife node show target1
```

```
Node Name:      target1
Environment:    _default
FQDN:          ip-10-154-155-107.ec2.internal
IP:            54.242.35.165
Run List:
Roles:
Recipes:
Platform:      ubuntu 12.04
Tags:
```

What is the Node object

- Nodes are made up of Attributes
 - Many are discovered **automatically** (platform, ip address, number of CPUs)
 - Many other objects in Chef can also add Node attributes (Cookbooks, Roles and Environments, Recipes, Attribute Files)
- Nodes are stored and indexed on the Chef Server

Exercise: Run Ohai on node

```
opscode@target1:~$ sudo ohai | less
```

```
{
  "languages": {
    "ruby": {

    },
    "python": {
      "version": "2.7.3",
      "builddate": "Apr 10 2013, 06:20:15"
    },
    "perl": {
      "version": "5.14.2",
      "archname": "x86_64-linux-gnu-thread-multi"
    }
  },
  "kernel": {
```

Exercise: Show all the node attributes

```
$ knife node show target1 -l
```

```
Node Name:    target1
Environment:  _default
FQDN:        ip-10-154-155-107.ec2.internal
IP:          54.242.35.165
Run List:
Roles:
Recipes:
Platform:    ubuntu 12.04
Tags:
Attributes:
tags:

Default Attributes:

Override Attributes:

Automatic Attributes (Ohai Data):
block_device:
loop0:
removable: 0
size:      0
```

Exercise: Show the raw node object

```
$ knife node show target1 -Fj
```

```
{
  "name": "target1",
  "chef_environment": "_default",
  "run_list": [],
  "normal": {"tags":[]}
}
```

Exercise: Show only the fqdn attribute

```
$ knife node show target1 -a fqdn
```

```
target1:  
fqdn: ip-10-154-155-107.ec2.internal
```

Exercise: Use search to find the same data

```
$ knife search node "*:*" -a fqdn
```

```
1 items found
```

```
target1:
```

```
fqdn: ip-10-154-155-107.ec2.internal
```

Templates and Cross-platform

Revisit the Apache Cookbook