ExcelShore®
Operational Excellence

Cybage®
Delivering Value. Scientifically.

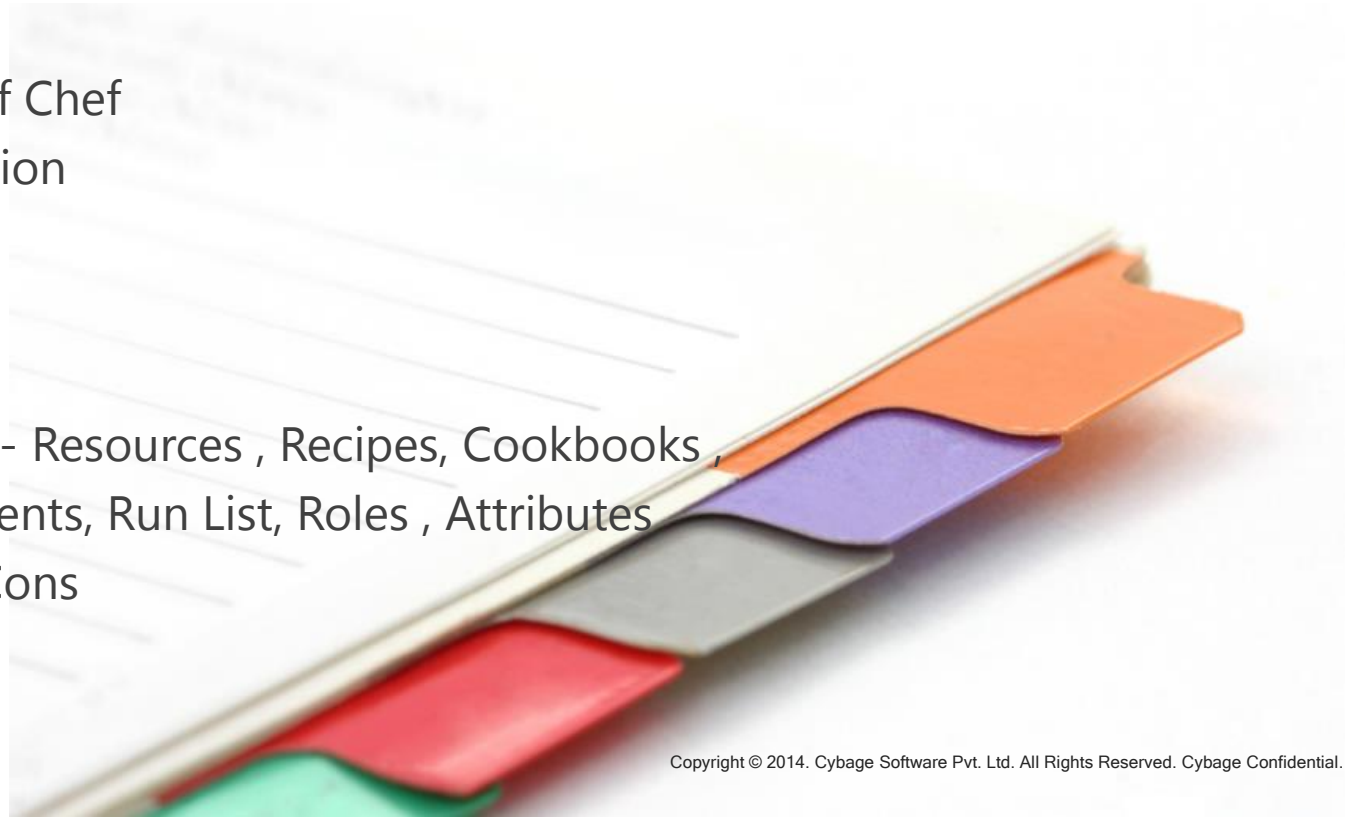
Welcome to Cybage

Authored and Presented by : Ismail Hatimbhai

Cybage®
Delivering Value. Scientifically.

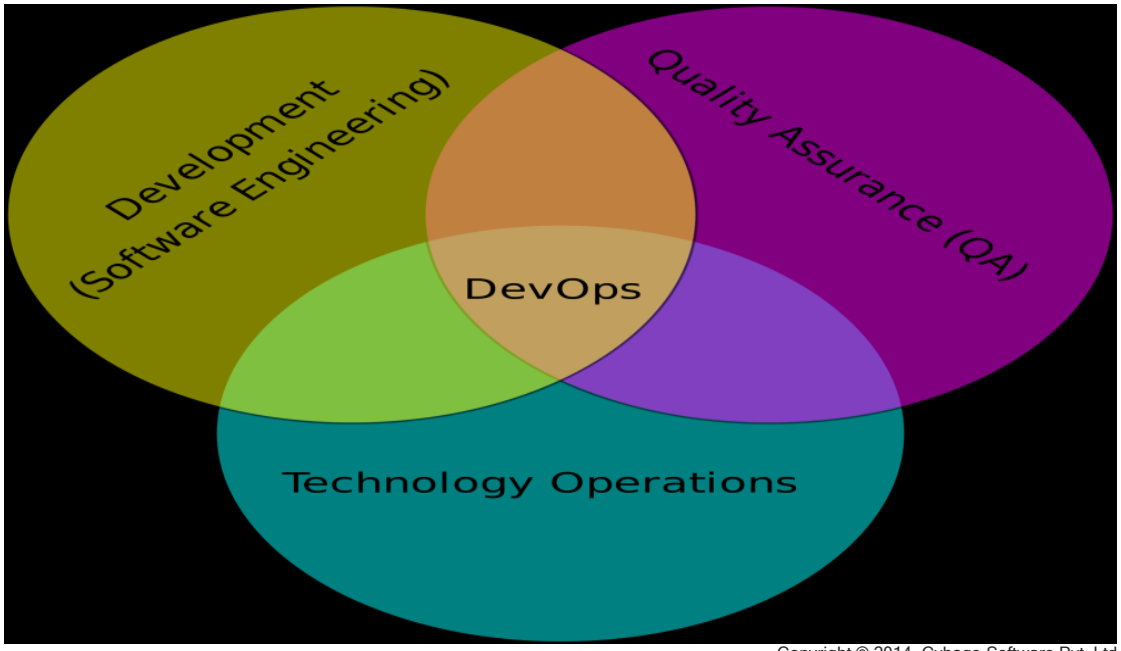
Agenda

- What is DevOps?
- Infrastructure As A Code
- Before and After Introduction of Chef
- History of Chef
- What is Chef
- Architecture of Chef
- Chef Workstation
- Chef Server
- Client Node
- Knife
- Key Concepts - Resources , Recipes, Cookbooks , Environments, Run List, Roles , Attributes
- Benefits and Cons
- References



DevOps

- Development + Operations = DevOps
- It's a concept dealing with software development, operations, and services.
- It emphasizes communication, collaboration, and integration between software developers and information technology (IT) operations personnel.



Infrastructure as a Code

IAAC:

- Configuration management
- IT management
- Provisioning
- Scripted infrastructures
- System configuration management
- and many other overlapping terms

Cybage® Delivering Value. Scientifically.

Before and After Chef

Before Chef

- Long and Tedious scripts
- Difficult to maintain
- Run scripts manually one machine-one script at a time
- Need lots of time
- Difficult to track
- Can not handle Hybrid Infrastructure
- Need Linux expert

After Chef

- Short and readable
- Easy to maintain
- Can run multiple scripts on 1000s of nodes at time
- Very less time
- Easy to track
- Easy way to handle Hybrid Infrastructure
- Basic Linux knowledge is sufficient

History of Chef

- Idea of Adam Jacob
- Reason was to find solution for System Integration Problems
- First released in 2009
- Community was built around the tool by OpsCode
- Chef was made Open Source

Cybage®
Delivering Value. Scientifically.

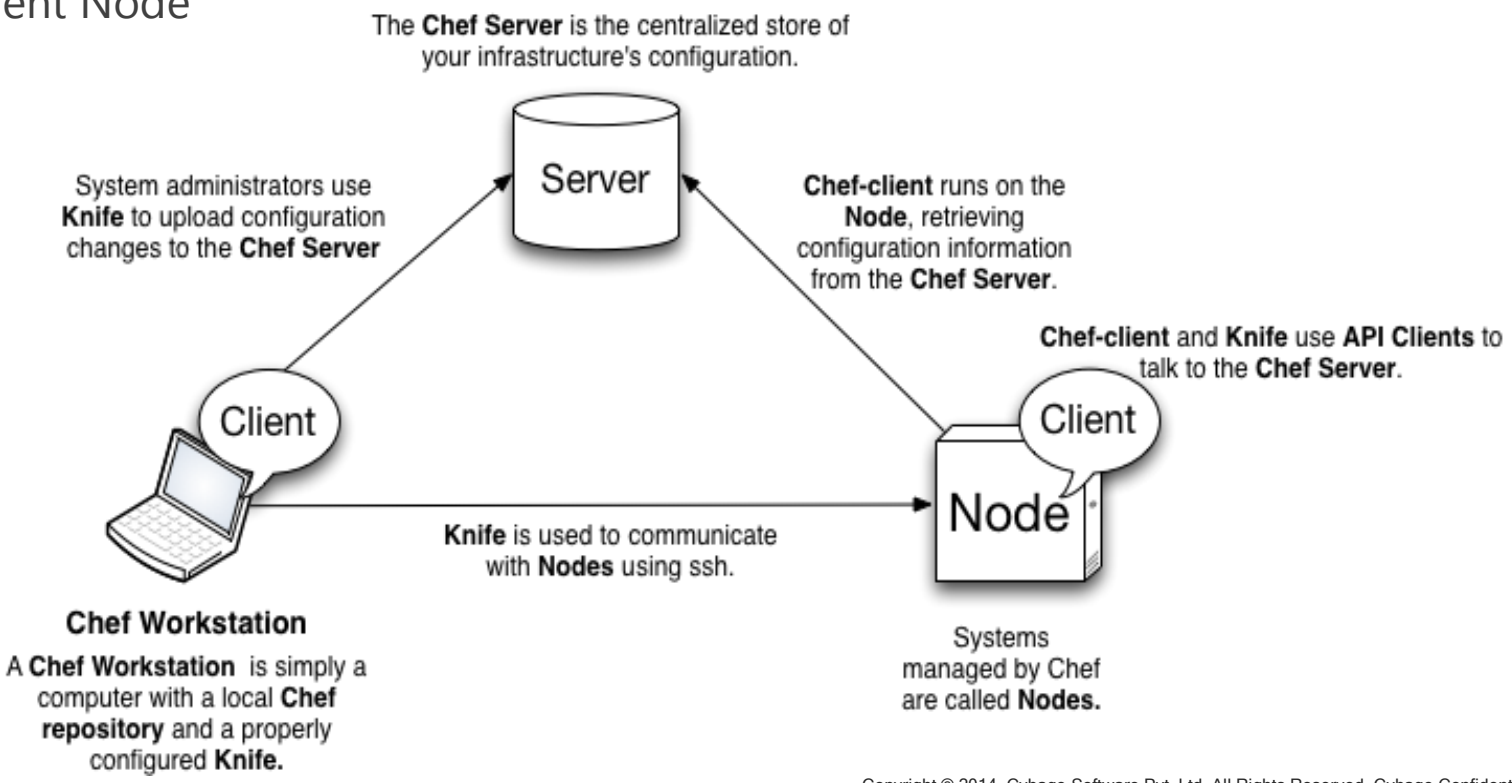
What is Chef?

- A configuration management tool
- A library/framework for configuration management
- Open Source! (Apache License, version 2.0)
- Building and managing infrastructure as a program

Architecture of Chef

The architecture consists of

- Chef Workstation
- Chef Server
- Client Node



Chef Workstation

- Place to create the configuration management code
- Place from where the code will be uploaded/download to the chef server
- Tells server which machine is to be configured
- Place where you will keep the **Chef Development Kit** (chef-repo)

Chef Server

- Storage of our code
- Hub for uploading and Downloading your Chef Cookbooks
- Keeps track of the status of all the nodes present in the network
- Available in three forms
  - Hosted/Enterprise Chef – Hosted by OpsCode
  - Private Hosted Chef – Hosted by OpsCode but from within the firewall
  - Open Source Chef – This is the Open Source version

Client Node

- Machine where the actual installation of resources would take place
- This machine needs to have **chef-client** in it.
- If not then install it using Chef Workstation
- If workstation is directly connected then it is **chef-solo** architecture
- It could be a bare machine or a machine on cloud

Knife

- Utility provided by Chef to communicate with Server and Client
- It helps
  - To configure a node
  - To upload or download anything from the chef server
  - To delete something (RISKY)
  - To get update of the chef server
  - To ask client to give status
  - To download something from supermarket of chef (Dangerously handy place) , etc.

Cybage®
Delivering Value. Scientifically.

Key Concepts

- Resources
- Recipes
- Cookbooks
- Environments
- Run List
- Roles
- Attributes

Cybage® Delivering Value. Scientifically.

Resources

- Resources are specified in recipes, recipes stored in cookbooks.
- The expanded run list species all the recipes (and thus the resources) to manage on a given node.
- A resource
  - has a type
  - has a name
  - has parameters
  - takes actions

```
package "tar" do
    version "1.16.1-1"
    action :install
end
```

- Actions are taken using **providers**, providers are chosen based on the node platform.
- (i.e. the package resource installs packages using apt on debian/ubuntu and using yum on centos/RHEL)!

Recipes

- This can be said as the most important part of Chef
- Ruby files that are saved here consists of the actual configuration management code
- Recipes evaluate resources in the order they appear

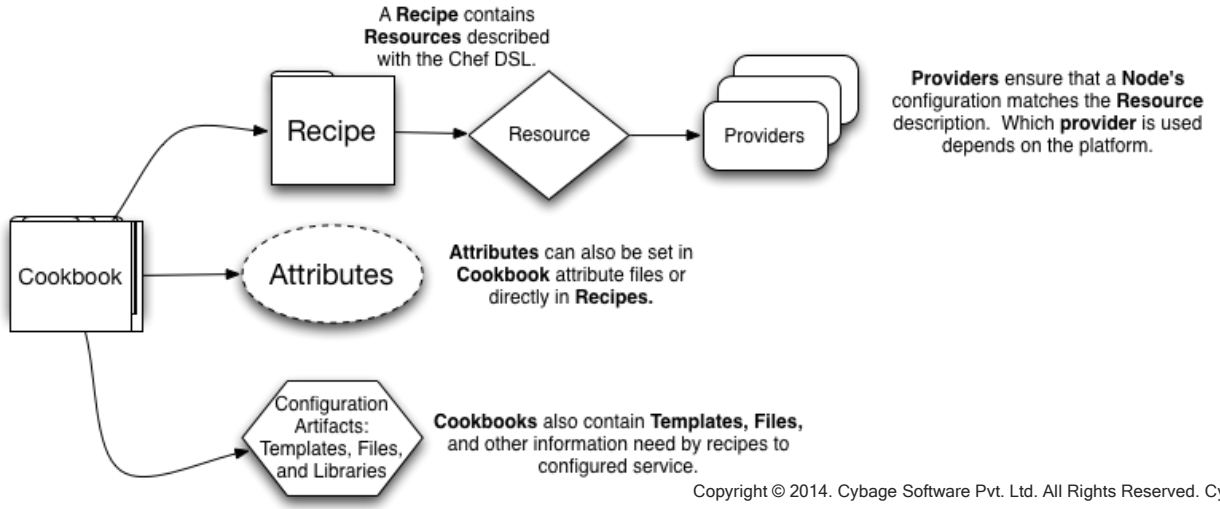
```
package 'httpd'

service 'httpd' do
        action [:start, :enable]
end

template '/var/www/html/index.html' do
        source 'index.html.erb'
end

service 'iptables' do
        action :stop
end
```

- Recipes can include other resources, and are just ruby code

```
include_recipe “mysql"
```

Cookbooks

- Made up of recipe, attributes, files, templates
- Used to organize the code
- This along with roles(if present) are what gets uploaded to the server
- Commands related to a cookbook
  - *knife cookbook create COOKBOOK_NAME*
  - *knife cookbook delete COOKBOOK_NAME*
  - *knife cookbook download COOKBOOK_NAME*
  - *knife cookbook list*



Cybage® Delivering Value. Scientifically.

Environments

- By default Environments folder is not created
- We can create it in many ways
- Simplest – Create a environment folder in chef-repo
- Environments can be used to manage different environments (production, test, etc) in a single Chef setup.
- Roles can have different run list on different environments

```
name "production"
description "The production environment"
cookbook_versions(
"mysql" => "= 1.2.5", # use version 1.2.5 only
"apache2" => "~> 1.1" # anything 1.1.0 < x < 1.2.0
)
# default attributes for this environment
attributes(
"apache2" => {
"listen_ports" => ["80", "443"]
}
)
```

- The ruby DSL gets compiled to JSON when uploading to server

Cybage® Delivering Value. Scientifically.

Run List

- Run list is a block of code which includes a number of recipes
- If there is need of running multiple recipes on a single machine **run_list** is included in the **Roles**

```
"run_list": {
   "role[python_hosting]",
   "recipe[postgresql::client]",
   "recipe[chishop]"
}
```

Roles

- Have attributes
- Have a run list
- Declared in JSON or . . .
- Declared with the ruby DSL (automatically compiled to JSON)
- An example role (in ruby):

```
name "python_hosting"
description "Python App hosting"
default_attributes(
        "nginx" => {
        "default_site_enabled" => false
        }
)
run_list(
        "recipe[python::virtualenv]",
        "recipe[uwsgi]",
        "recipe[nginx]"
)
```

Attributes

- Store node data (i.e. ip address, hostname, fqdn, database host address, etc.)
- There are four types of attributes (in order of precedence, lowest to highest):
  - Default
  - Normal
  - Override
  - Automatic
- Attributes can be set in:
  - Cookbooks
  - Environments
  - Roles
  - Nodes

Attributes (cont...)

- As attributes are deep-merged, the following precedence applies
  - **default** attributes applied in an **cookbook**
  - **default** attributes applied in an **environment**
  - **default** attributes applied in a **role**
  - **default** attributes applied on a **node** directly in a **recipe**
  - **normal** attributes applied in a **cookbook**
  - **normal** attributes applied on a **node** directly in a **recipe**
  - **override** attributes applied in an **cookbook**
  - **override** attributes applied in an **environment**
  - **override** attributes applied in a **role**
  - **override** attributes applied on a **node** directly in a **recipe**
  - **automatic** attributes generated by **Ohai**

- Automatic, override and default are reset at the beginning of every run.
- Normal attributes persist between runs.

Benefits and Cons

Pros

- Infrastructure is in the form of code
- Use of High level DSL, making understanding of code easy
- Ohai – collects all the data on the node and provides it while running, saving a lot of time
- Alone knife tool capable of doing most of your admin work
- Being in Ruby – you don't need to know complete ruby just basics would be fine.

Cons

- Documentation is very scattered and difficult to understand
- Error log very complex which makes it difficult to resolve
- For testing **Test-Kitchen** is available but proper example use is not given
- Is dependent on Ruby

Cybage®
Delivering Value. Scientifically.

References

- http://opscode.com/
- http://community.opscode.com/
- http://docs.opscode.com/
- http://learnchef.com
- http://lists.opscode.com
- http://youtube.com/user/Opscode

Any Questions?

ExcelShore®
Operational Excellence

Cybage®
Delivering Value. Scientifically.

Thank You!