

Data Bags

Working with Custom Data Sets

Objectives



After completing this module, you should be able to

- Explain how to use and manage a Data Bag
- Create and upload a Data Bag to Chef Server
- Query Data Bag information with the CLI
- Generate the 'myusers' cookbook
- Add the 'myusers' cookbook to company_web policyfile's run list
- Update and push the policyfile to the Chef Infra Server
- Converge web nodes

CONCEPT



E.g. User data or Env related data

Where should we store data that each node might need access to?

CONCEPT



Data Bags

A data bag is a container for items that represent information about your infrastructure that is **not tied to a single node**.

Examples:

- Users
- Groups
- Application Release Information
- Passwords (in an encrypted data bag)

https://docs.chef.io/data_bags.html

CONCEPT



Data Bags

- Data bags store global variables as JSON data.
- Data bags are indexed for searching and can be loaded by a cookbook or accessed during a search.
- Can be created manually or using knife

https://docs.chef.io/data_bags.html



Group Lab: Custom Data Sets

We can store sets of JSON data on our Chef Server, accessible by a node with search

Objective:

- ☐ Create “users” data bags
- ☐ Upload data bags to Chef Server
- ☐ Use CLI to query information about data bags

GL: What Can 'knife data bag' Do?



```
$ cd ~/chef-repo
```

```
$ knife data bag --help
```

```
** DATA BAG COMMANDS **
```

```
knife data bag create BAG [ITEM] (options)
```

```
knife data bag delete BAG [ITEM] (options)
```

```
knife data bag edit BAG ITEM (options)
```

```
knife data bag from file BAG FILE|FOLDER [FILE|FOLDER..] (options)
```

```
knife data bag list (options)
```

```
knife data bag show BAG [ITEM] (options)
```

GL: Run 'knife data bag list'



```
$ knife data bag list
```


GL: Create a data_bags Directory



Create directory to hold all data_bags in workstation

```
$ mkdir data_bags
```

Create directory to hold data bag {users} in workstation

```
$ mkdir data_bags/users
```

Create data bag {users} in server

```
$ knife data bag create users
```

GL: Create centos_user.json

 `~/chef-repo/data_bags/users/centos_user.json`

```
{  
  "id": "centos_user",  
  "comment": "I am a centos_user user",  
  "platform": "centos"  
}
```

GL: Create windows_user.json

 ~/chef-repo/data_bags/users/windows_user.json

```
{  
  "id": "windows_user",  
  "comment": "I am a windows user",  
  "platform": "windows"  
}
```



Group Lab: Custom Data Sets

We can store sets of JSON data on our Chef Server, accessible by a node with search

Objective:

- ✓ Create “users” data bags
- ❑ Upload data bags to Chef Server
- ❑ Use CLI to query information about data bags

GL: Upload Data Bag Items to Chef Server



```
$ knife data bag from file users data_bags/users/centos_user.json  
data_bags/users/windows_user.json
```

```
Updated data_bag_item[users::centos_user]  
Updated data_bag_item[users::windows_user]
```

You can also do `data_bags/users/*`

You should still be in `~/chef-repo` when running this command.

GL: Upload Data Bag Items to Chef Server



```
$ knife data bag from file myusers data_bags/users/centos_user.json
```

This data bag do not exist in server

```
ERROR: The object you are looking for could not be found  
Response: No data bag 'myusers' could be found. Please create this data bag  
before adding items to it.
```

GL: Validate Chef Server received items



```
$ knife data bag list
```

```
users
```

```
$ knife data bag show users
```

```
centos_user  
windows_user
```



Group Lab: Custom Data Sets

We can store sets of JSON data on our Chef Server, accessible by a node with search

Objective:

- ✓ Create “users” data bags
- ✓ Upload data bags to Chef Server
- Use CLI to query information about data bags

GL: View Details of centos_user



```
$ knife data bag show users centos_user
```

```
comment: I am a centos_user user  
id:      centos_user  
platform: centos
```

GL: Search the users Index



```
$ knife search users "*:*"
```

```
2 items found
```

```
chef_type: data_bag_item
comment:   I am a centos_user user
data_bag:  users
id:        centos_user
platform:  centos
```

```
chef_type: data_bag_item
comment:   I am a windows user
data_bag:  users
id:        windows_user
platform:  windows
```

The data bag {name} becomes SOLR index

GL: Return Users with “platform:centos”



```
$ knife search users "platform:windows"
```

```
1 items found
```

```
chef_type: data_bag_item  
comment:   I am a windows user  
data_bag:  users  
id:        windows_user  
platform:  windows
```



Group Lab: Custom Data Sets

We can store sets of JSON data on our Chef Server, accessible by a node with search

Objective:

- ✓ Create “users” data bags
- ✓ Upload data bags to Chef Server
- ✓ Use CLI to query information about data bags



GL: Create Users from a Data Bag

Dynamically search through the Chef Server under the 'users' index to create users

Objective:

- ☐ Generate the 'myusers' cookbook
- ☐ Create users based on data bag contents within default recipe
- ☐ Add the 'myusers' cookbook to company_web policyfile's run list
- ☐ Update and push the policyfile to the Chef Infra Server
- ☐ Converge web nodes

CONCEPT



Where are the users?

Because our users are now indexed on the Chef Server, we have a centralized source of truth for information regarding these users.

We can search through this information inside of our recipes.

GL: Generate the myusers Cookbook



```
$ cd ~/chef-repo
```

```
$ chef generate cookbook cookbooks/myusers
```

Generating cookbook myusers

- Ensuring correct cookbook content
- Committing cookbook files to git

Your cookbook is ready. To setup the pipeline, type `cd cookbooks/myusers`, then run `delivery init`



GL: Create Users from a Data Bag

Dynamically search through the Chef Server under the 'users' index to create users

Objective:

- ✓ Generate the 'myusers' cookbook
- ❑ Create users based on data bag contents within default recipe
- ❑ Add the 'myusers' cookbook to company_web policyfile's run list
- ❑ Update and push the policyfile to the Chef Infra Server
- ❑ Converge web nodes

GL: Create users recipe

 `~/chef-repo/cookbooks/myusers/recipes/users.rb`

```
system_users = search("users", "platform:#{node['platform']}")
```

```
system_users.each do |user_data|
```

```
  user user_data['id'] do
```

```
    comment user_data['comment']
```

```
    action :create
```

```
  end
```

```
end
```

Gets {ohai platform} for the machine where it is running and queries for data bags. For centos VMs, this becomes platform:centos. For Win VMs, this becomes platform:windows

Thus, users defined for each platform will be created using data bag appropriately

Using {user} resource to create users

GL: Include users recipe within default recipe

 `~/chef-repo/cookbooks/myusers/recipes/default.rb`

```
#  
# Cookbook:: myusers  
# Recipe:: default  
#  
# Copyright:: 2018, The Authors, All Rights Reserved.  
  
include_recipe 'myusers::users'
```



GL: Create Users from a Data Bag

Dynamically search through the Chef Server under the 'users' index to create users

Objective:

- ✓ Generate the 'myusers' cookbook
- ✓ Create users based on data bag contents within default recipe
- ❑ Add the 'myusers' cookbook to company_web policyfile's run list
- ❑ Update and push the policyfile to the Chef Infra Server
- ❑ Converge web nodes

GL: Change to the cookbooks/myusers Directory



```
$ cd cookbooks/myusers
```

GL: Add myusers Cookbook to the company_web Policyfile

```
~/chef-repo/company_web.rb
```

```
...skipping...
```

```
# run_list: chef-client will run these recipes in the order specified.
```

```
run_list 'company_web::default', 'myusers::default'
```

```
# Specify a custom source for a single cookbook:
```

```
cookbook 'company_web', path: 'cookbooks/company_web'
```

```
cookbook 'myiis', path: 'cookbooks/myiis'
```

```
cookbook 'apache', path: 'cookbooks/apache'
```

```
cookbook 'myusers', path: 'cookbooks/myusers'
```



GL: Create Users from a Data Bag

Dynamically search through the Chef Server under the 'users' index to create users

Objective:

- ✓ Generate the 'myusers' cookbook
- ✓ Create users based on data bag contents within default recipe
- ✓ Add the 'myusers' cookbook to company_web policyfile's run list
- ❑ Update and push the policyfile to the Chef Infra Server
- ❑ Converge web nodes

GL: Ensure You are in chef-repo



```
$ cd ~/chef-repo
```

GL: Update the Policyfile



```
$ chef update company_web.rb
```

```
Attributes already up to date
Building policy company_web
Expanded run list: recipe[mychef_client::default], recipe[company_web::default],
recipe[myusers::default]
Caching Cookbooks...
Installing company_web      >= 0.0.0 from path
Installing myiis            >= 0.0.0 from path
Installing apache          >= 0.0.0 from path
Installing mychef_client    >= 0.0.0 from path
Installing myusers          >= 0.0.0 from path
Using      chef-client      11.2.0
Using      cron             6.2.1
Using      logrotate        2.2.0

Lockfile written to /Users/sdelfante/chef-repo/company_web.lock.json
Policy revision id: 14a342b5d2134516984b878c1f6be83a51427e6f84594c4dd9e57456b1312582
```


GL: Push the Policyfile to Chef Infra Server



```
$ chef push prod company_web.lock.json
```

```
Uploading policy company_web (14a342b5d2) to policy group prod
```

```
Using      apache      0.1.0      (1388ab3a)
```

```
Using      chef-client  11.2.0   (0b49a3a8)
```

```
Using      company_web  0.1.0    (c1b26cb5)
```

```
Using      cron         6.2.1    (08676b5c)
```

```
Using      logrotate    2.2.0    (53e09234)
```

```
Using      mychef_client 0.1.0    (10d082a4)
```

```
Using      myiis        0.2.1    (cd0db3ed)
```

```
Uploaded myusers        0.1.0    (bebccee3)
```

Note that:
new policy is uploaded to prod only



GL: Create Users from a Data Bag

Dynamically search through the Chef Server under the 'users' index to create users

Objective:

- ✓ Generate the 'myusers' cookbook
- ✓ Create users based on data bag contents within default recipe
- ✓ Add the 'myusers' cookbook to company_web policyfile's run list
- ✓ Update and push the policyfile to the Chef Infra Server
- ❑ Converge web nodes

GL: Converge All Web Nodes



```
$ knife ssh "policy_name:company_web" -x centos -i ~/aws.pem "sudo chef-client"
```

```
$ knife winrm "name:iis_web" -a cloud.public_ipv4 -x USER -P PWD "chef-client"
```

GL: Check Local Users for Apache Server



```
$ knife ssh "policy_name:company_web AND policy_group:prod" -x  
centos -i ~/aws.pem "cat /etc/passwd"
```

```
ec2-52-15-221-52.us-east-2.compute.amazonaws.com  
postfix:x:89:89::/var/spool/postfix:/sbin/nologin  
ec2-52-15-221-52.us-east-2.compute.amazonaws.com  
chrony:x:998:995::/var/lib/chrony:/sbin/nologin  
ec2-52-15-221-52.us-east-2.compute.amazonaws.com  
centos:x:1000:1000:Cloud User:/home/centos:/bin/bash  
ec2-52-15-221-52.us-east-2.compute.amazonaws.com  
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin  
ec2-52-15-221-52.us-east-2.compute.amazonaws.com  
centos_user:x:1001:1001:I am a centos_user  
user:/home/centos_user:/bin/bash
```

```
> knife winrm "name:iis_web" -a cloud.public_ipv4 -x USER -P PWD "net user windows_user"
```

GL: Check Local Users for Apache Server



```
$ knife ssh "policy_name:company_web AND policy_group:acceptance" -  
x centos -i ~/aws.pem "cat /etc/passwd"
```

You won't find centos_user

This will happen because in the last chapter, we changed one of the webserver node to {acceptance} policy_group, and we applied the new {company_web} policy only to {prod} policy_group



GL: Create Users from a Data Bag

Dynamically search through the Chef Server under the 'users' index to create users

Objective:

- ✓ Generate the 'myusers' cookbook
- ✓ Create users based on data bag contents within default recipe
- ✓ Add the 'myusers' cookbook to company_web policyfile's run list
- ✓ Update and push the policyfile to the Chef Infra Server
- ✓ Converge web nodes



Review Questions

1. When should we utilize data bags instead of node attributes?
2. When creating a new data bag, what index on the Chef server does the data bag get added to?



Q&A

What questions can we help you answer?



CHEF™