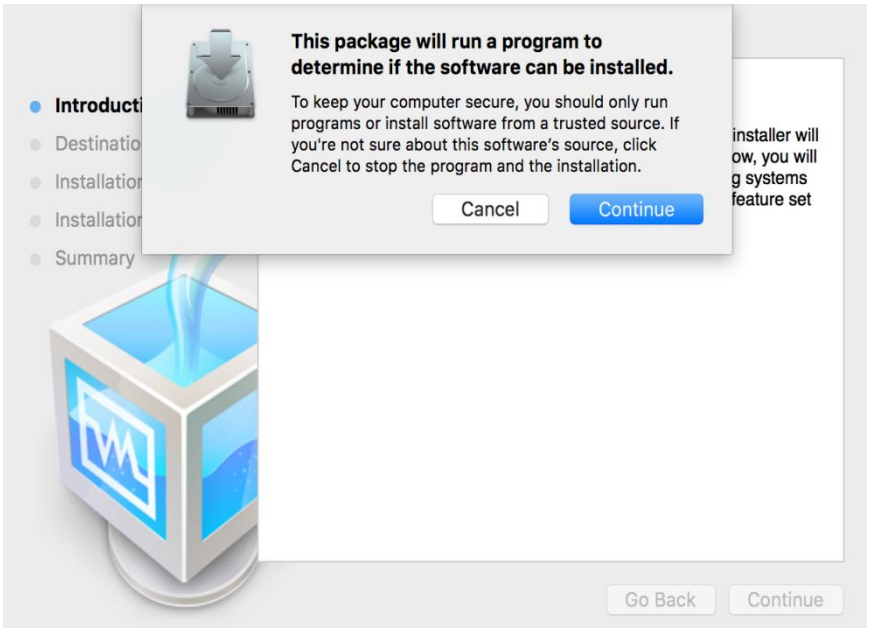
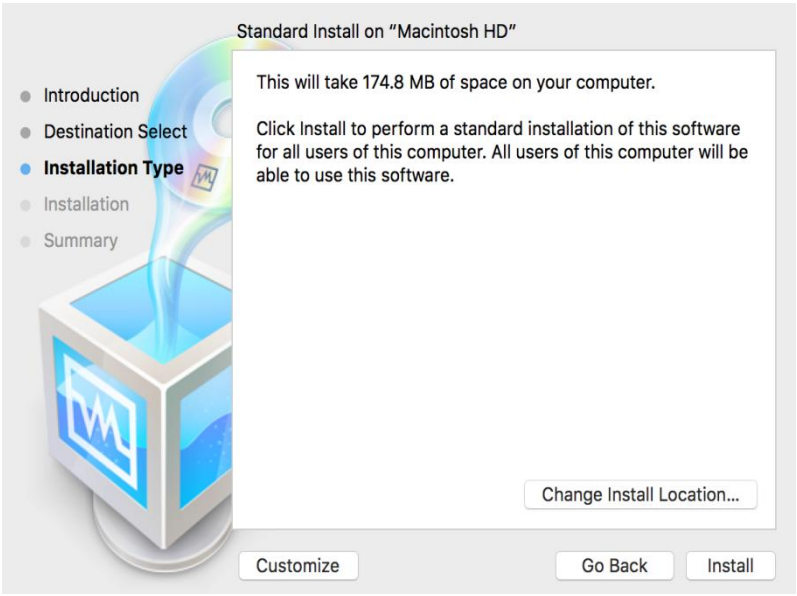
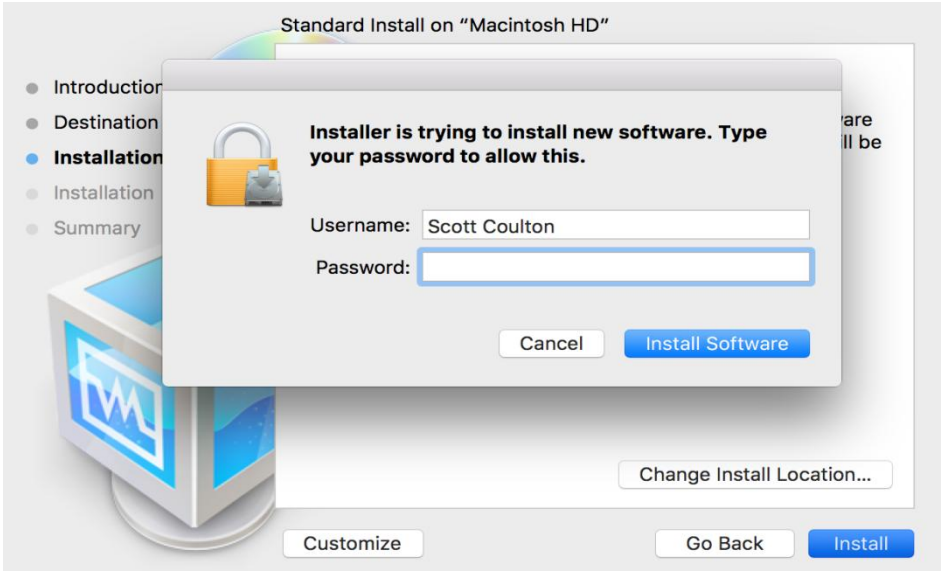
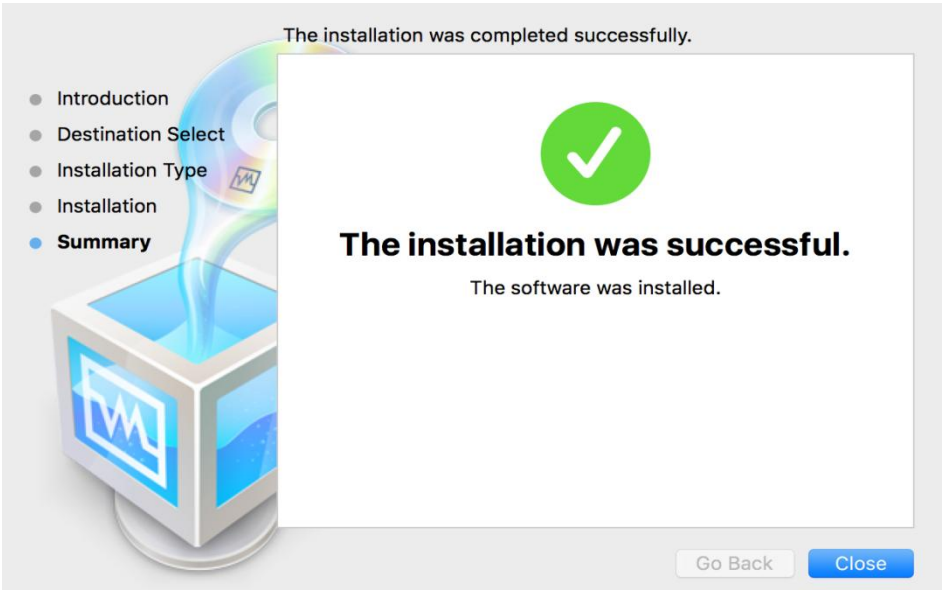
Chapter 1: Installing Docker with Puppet


This package will run a program to determine if the software can be installed.
To keep your computer secure, you should only run programs or install software from a trusted source. If you're not sure about this software's source, click Cancel to stop the program and the installation.
Cancel
Continue
Introducti
Destinatio
Installatio
Installatio
Summary
installer will
ow, you will
g systems
feature set
Go Back
Continue


Standard Install on "Macintosh HD"
Introduction
Destination Select
Installation Type
Installation
Summary
This will take 174.8 MB of space on your computer.
Click Install to perform a standard installation of this software for all users of this computer. All users of this computer will be able to use this software.
Change Install Location...
Customize
Go Back
Install


Standard Install on “Macintosh HD”
Introductior
Destination
Installation
Installation
Summary
Installer is trying to install new software. Type your password to allow this.
Username: Scott Coulton
Password:
Cancel
Install Software
Change Install Location...
Customize
Go Back
Install


The installation was completed successfully.
Introduction
Destination Select
Installation Type
Installation
Summary
The installation was successful.
The software was installed.
Go Back
Close

DOWNLOAD VAGRANT

Below are the available downloads for the latest version of Vagrant (1.7.4). Please download the proper package for your operating system and architecture.

You can find the SHA256 checksums for Vagrant 1.7.4 online and you can verify the checksums signature file which has been signed using HashiCorp's GPG key. You can also download older versions of Vagrant from the releases service.

MAC OS X

Universal (32 and 64-bit)

WINDOWS

Universal (32 and 64-bit)

DEBIAN

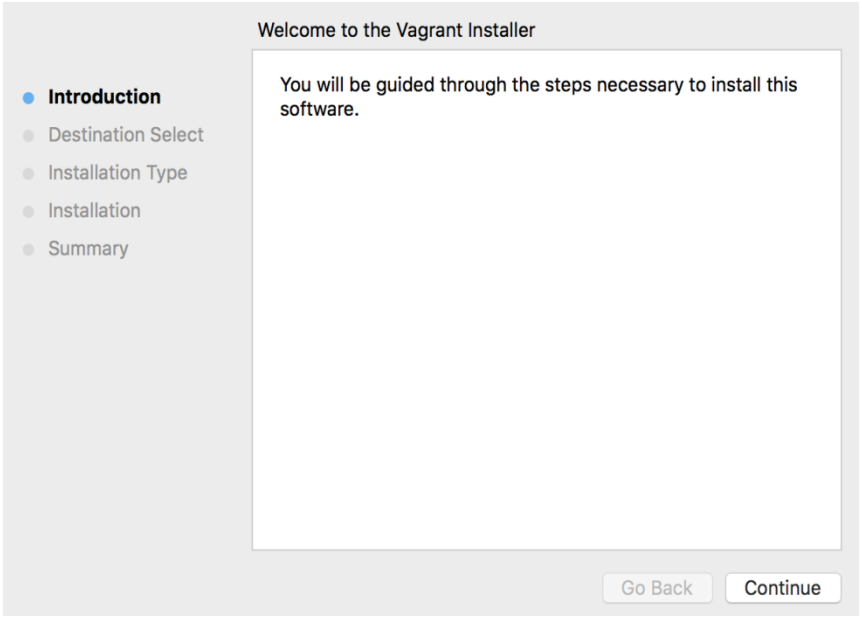
32-bit | 64-bit

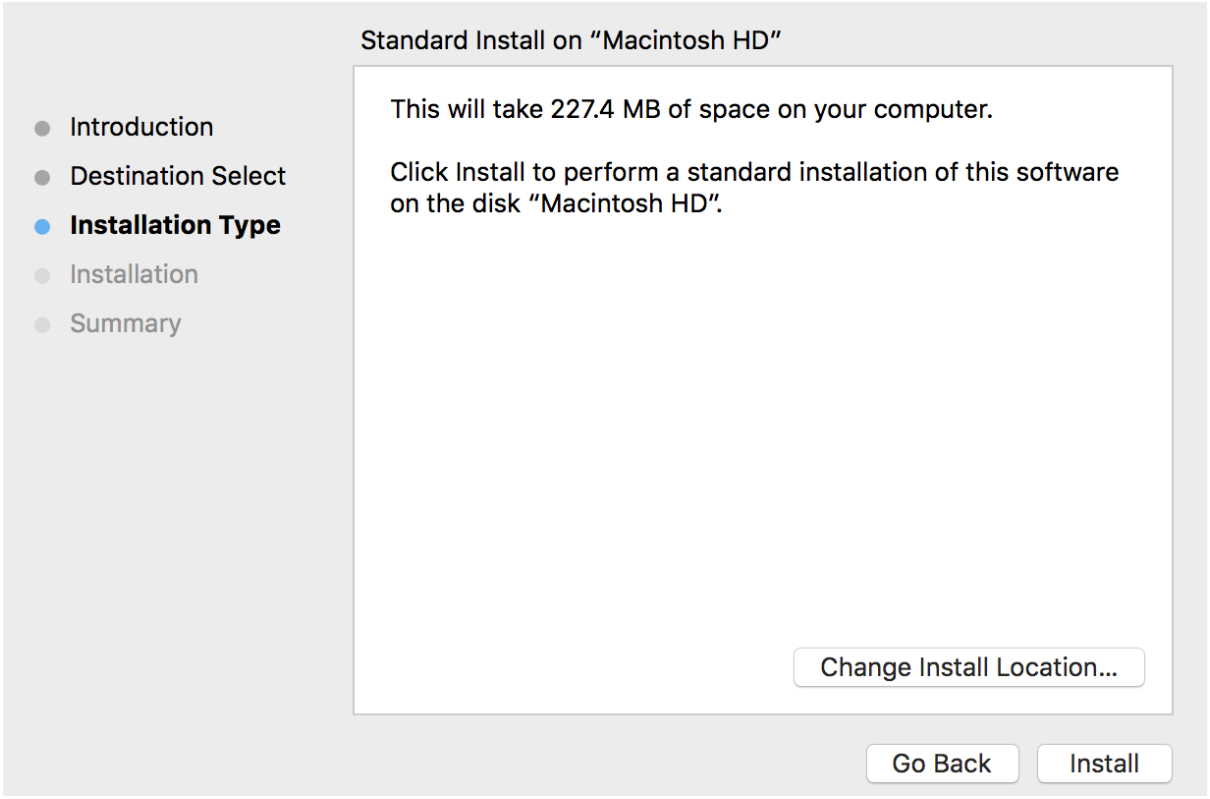
CENTOS

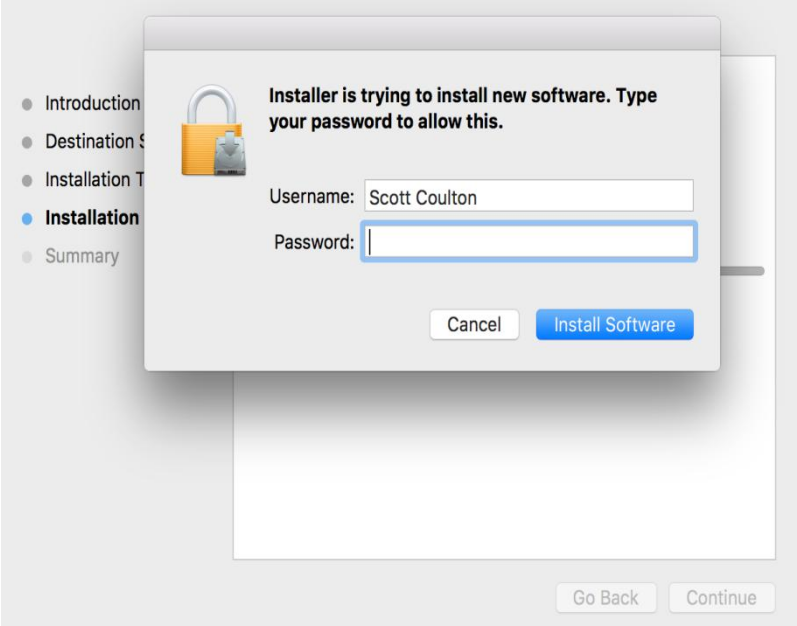
32-bit | 64-bit

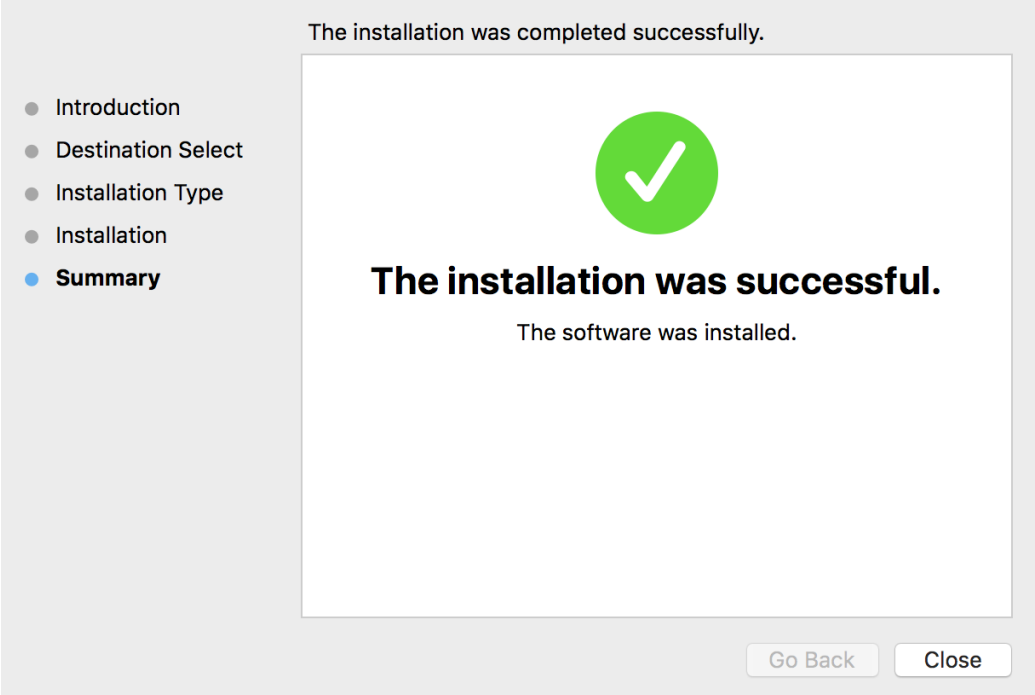
Vagrant.pkg

uninstall.tool




Standard Install on “Macintosh HD”
Introduction
Destination Select
Installation Type
Installation
Summary
This will take 227.4 MB of space on your computer.
Click Install to perform a standard installation of this software on the disk “Macintosh HD”.
Change Install Location...
Go Back
Install

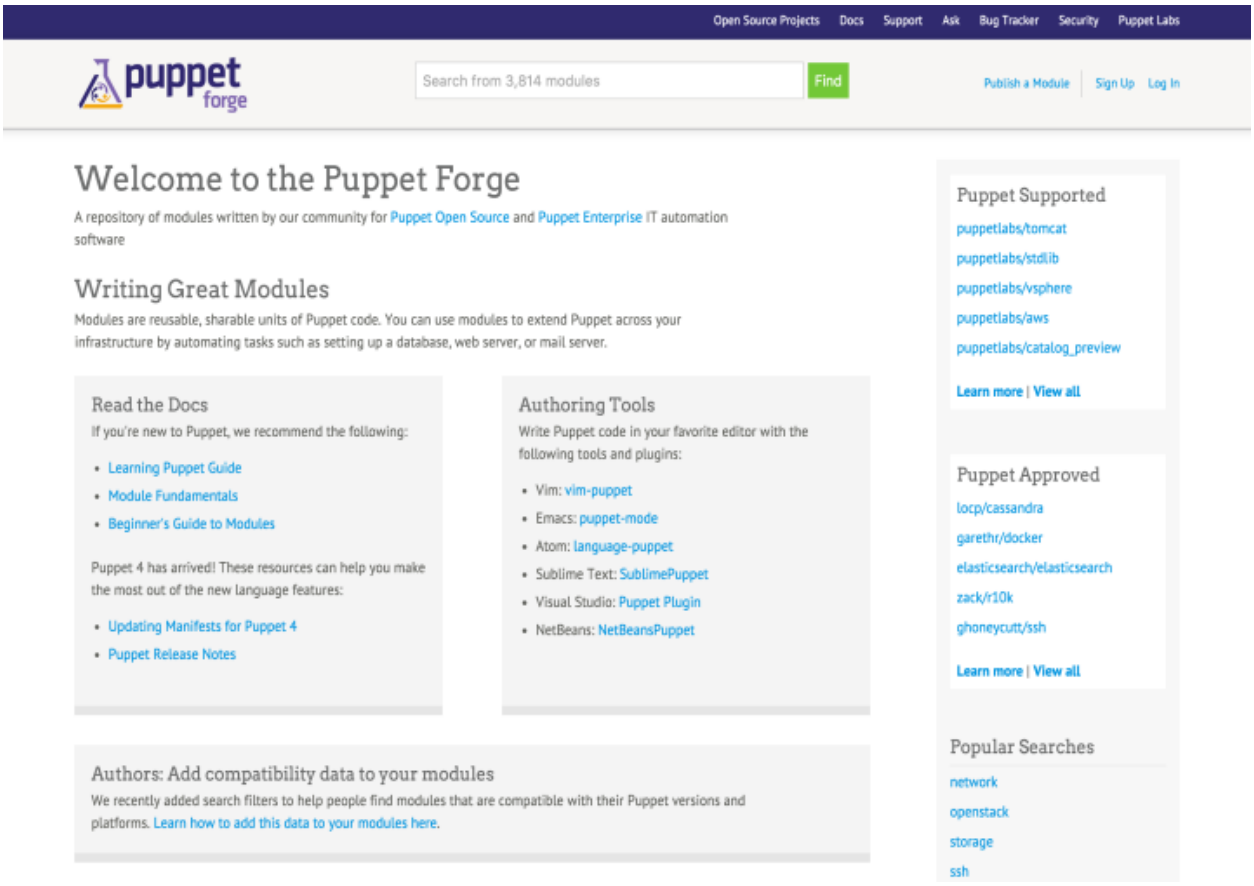

Introduction
Destination S
Installation T
Installation
Summary
Installer is trying to install new software. Type your password to allow this.
Username: Scott Coulton
Password:
Cancel
Install Software
Go Back
Continue

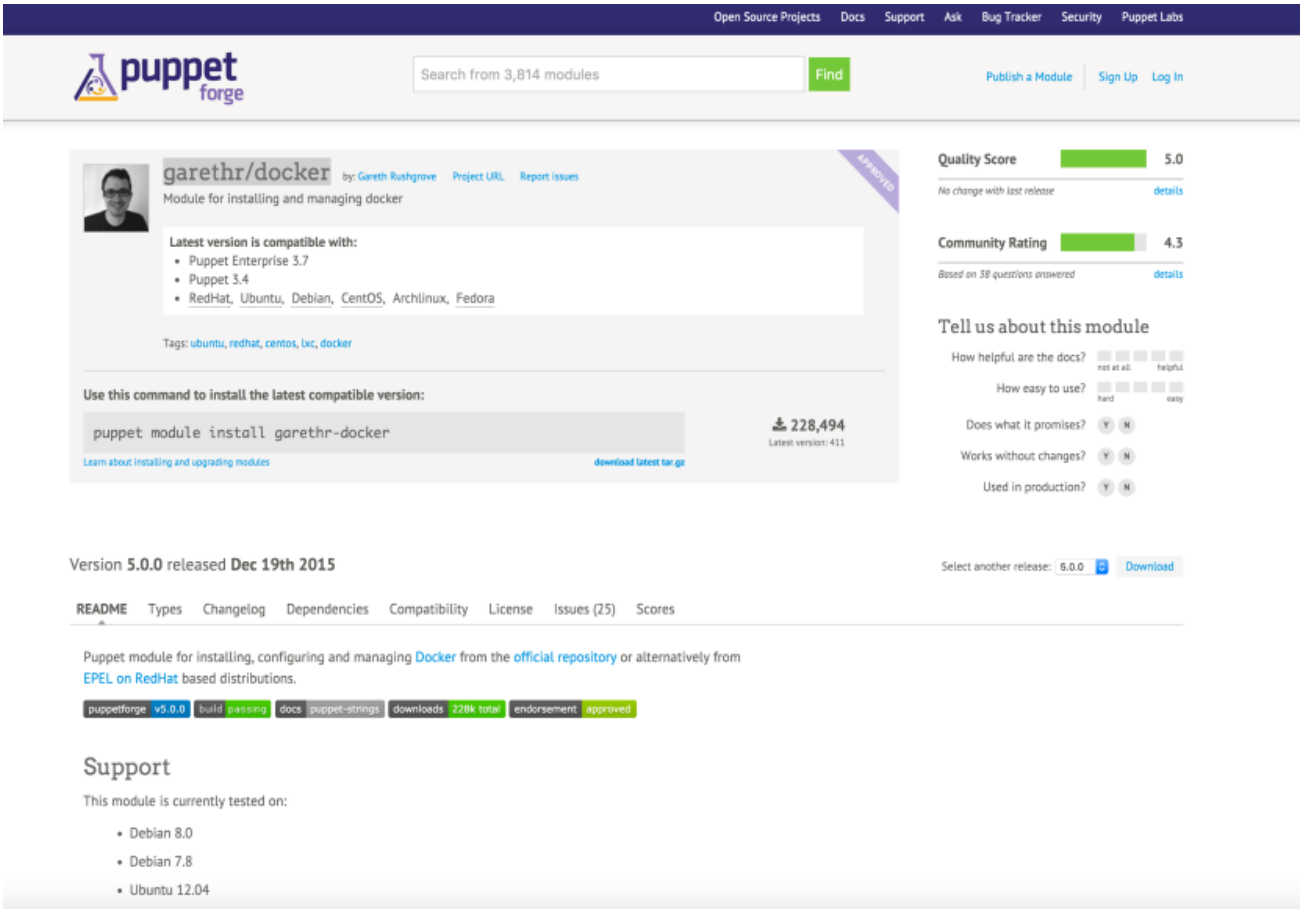

The installation was completed successfully.
Introduction
Destination Select
Installation Type
Installation
Summary
The installation was successful.
The software was installed.
Go Back
Close

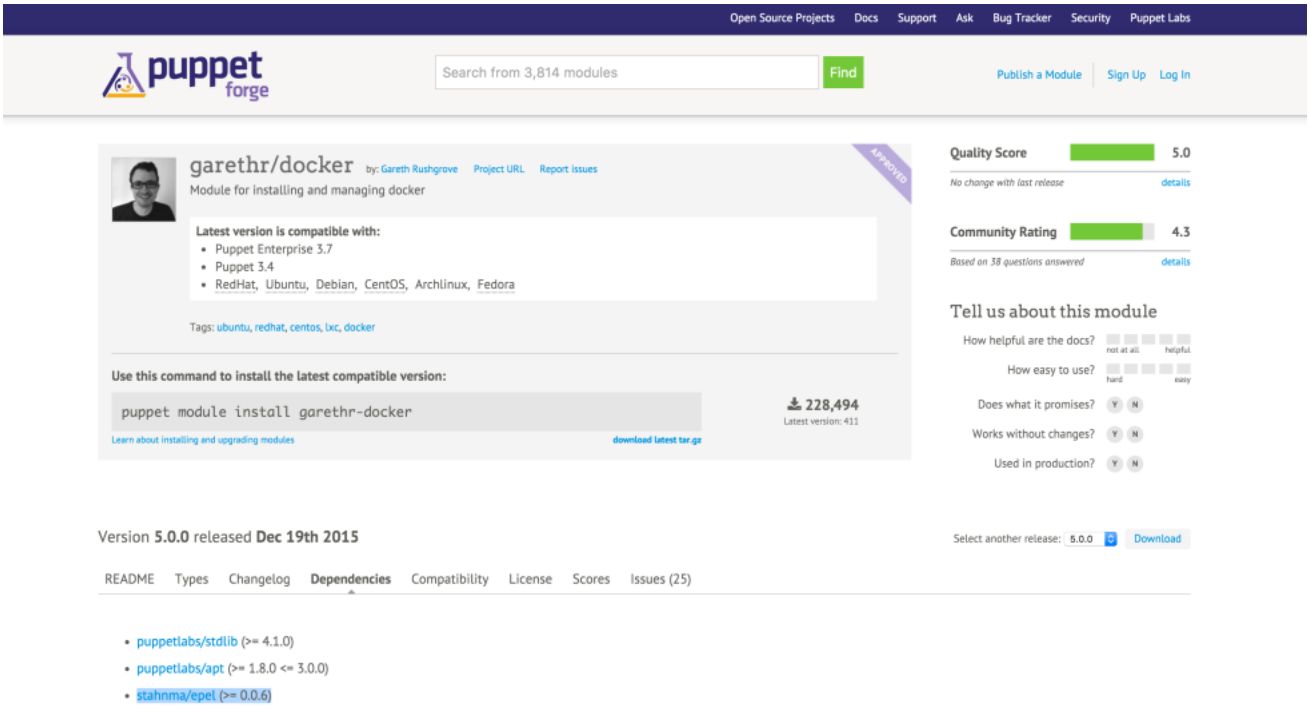
```
✔ [scottcoulton@Scotts-MacBook-Pro] ~
08:18 $ vagrant
Usage: vagrant [options] <command> [<args>]

    -v, --version                    Print the version and exit.
    -h, --help                       Print this help.

Common commands:
     box             manages boxes: installation, removal, etc.
     connect         connect to a remotely shared Vagrant environment
     destroy         stops and deletes all traces of the vagrant machine
     global-status   outputs status Vagrant environments for this user
     halt            stops the vagrant machine
     help            shows the help for a subcommand
     hosts           Information about hostnames managed by the vagrant-hosts plugin
     init            initializes a new Vagrant environment by creating a Vagrantfile
     login           log in to HashiCorp's Atlas
     oscar
     package         packages a running vagrant environment into a box
     pe-build        Commands related to PE Installation
     plugin          manages plugins: install, uninstall, update, etc.
     provision       provisions the vagrant machine
     push            deploys code in this environment to a configured destination
     rdp             connects to machine via RDP
     reload          restarts vagrant machine, loads new Vagrantfile configuration
     resume          resume a suspended vagrant machine
     scp             copies data into a box via SCP
     share           share your Vagrant environment with anyone in the world
     ssh             connects to machine via SSH
     ssh-config      outputs OpenSSH valid configuration to connect to the machine
     status          outputs status of the vagrant machine
     suspend         suspends the machine
     up              starts and provisions the vagrant environment
     vbguest
     version         prints current and latest Vagrant version

For help on any individual command run `vagrant COMMAND -h`

Additional subcommands are available, but are either more advanced
or not commonly used. To see all subcommands, run the command
`vagrant list-commands`.
```

```ruby
# -*- mode: ruby -*-
# # vi: set ft=ruby :

# Specify minimum Vagrant version and Vagrant API version
Vagrant.require_version ">= 1.6.0"
VAGRANTFILE_API_VERSION = "2"

# Require YAML module
require 'yaml'

# Read YAML file with box details
servers = YAML.load_file('servers.yaml')

# Create boxes
Vagrant.configure(VAGRANTFILE_API_VERSION) do |config|

  # Iterate through entries in YAML file
servers.each do |servers|

  config.vm.define servers["name"] do |srv|

    srv.vm.hostname = servers["name"]

    srv.vm.box = servers["box"]

    srv.vm.network "private_network", ip: servers["ip"]


   servers["forward_ports"].each do |port|
      srv.vm.network :forwarded_port, guest: port["guest"], host: port["host"]
  end

   srv.vm.provider :virtualbox do |v|
        v.cpus = servers["cpu"]
        v.memory = servers["ram"]
  end

    srv.vm.synced_folder "./", "/home/vagrant/#{servers['name']}"

    servers["shell_commands"].each do |sh|
      srv.vm.provision "shell", inline: sh["shell"]
    end

    srv.vm.provision :puppet do |puppet|
        puppet.temp_dir = "/tmp"
        puppet.options = ['--modulepath=/tmp/modules', '--verbose']
        puppet.hiera_config_path = "hiera.yaml"

        end
      end
    end
  end
```

```
---
-
  box: puppetlabs/centos-7.0-64-puppet-enterprise
  cpu: 1
  ip: "172.17.8.101"
  name: node-01
  forward_ports:
    - { guest: 80, host: 8080 }
  ram: 2048
  shell_commands:
    - { shell: 'yum install -y wget git lvm2 device-mapper-libs' }
    - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
    - { shell: 'cp /home/vagrant/node-01/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
```




Open Source Projects Docs Support Ask Bug Tracker Security Puppet Labs
puppet forge
Search from 3,814 modules
Find
Publish a Module Sign Up Log In
garethr/docker by: Gareth Rushgrove Project URL Report issues
Module for installing and managing docker
APPROVED
Latest version is compatible with:
Puppet Enterprise 3.7
Puppet 3.4
RedHat, Ubuntu, Debian, CentOS, Archlinux, Fedora
Tags: ubuntu, redhat, centos, lxc, docker
Use this command to install the latest compatible version:
puppet module install garethr-docker
228,494
Latest version: 411
Learn about installing and upgrading modules
download latest tar.gz
Quality Score 5.0
No change with last release details
Community Rating 4.3
Based on 38 questions answered details
Tell us about this module
How helpful are the docs?
not at all helpful
How easy to use?
hard easy
Does what it promises? Y N
Works without changes? Y N
Used in production? Y N
Version 5.0.0 released Dec 19th 2015
Select another release: 5.0.0
Download
README Types Changelog Dependencies Compatibility License Issues (25) Scores
Puppet module for installing, configuring and managing Docker from the official repository or alternatively from EPEL on RedHat based distributions.
puppetforge v5.0.0 build passing docs puppet-strings downloads 228k total endorsement approved
Support
This module is currently tested on:
Debian 8.0
Debian 7.8
Ubuntu 12.04




```
#!/usr/bin/ruby env

require "socket"
$hostname = Socket.gethostname

forge 'http://forge.puppetlabs.com'


mod 'puppetlabs/stdlib', '4.1.0'
mod 'puppetlabs/apt', '2.2.1'
mod 'stahnma/epel', '0.0.6'
mod 'garethr/docker', '5.0.0'
```

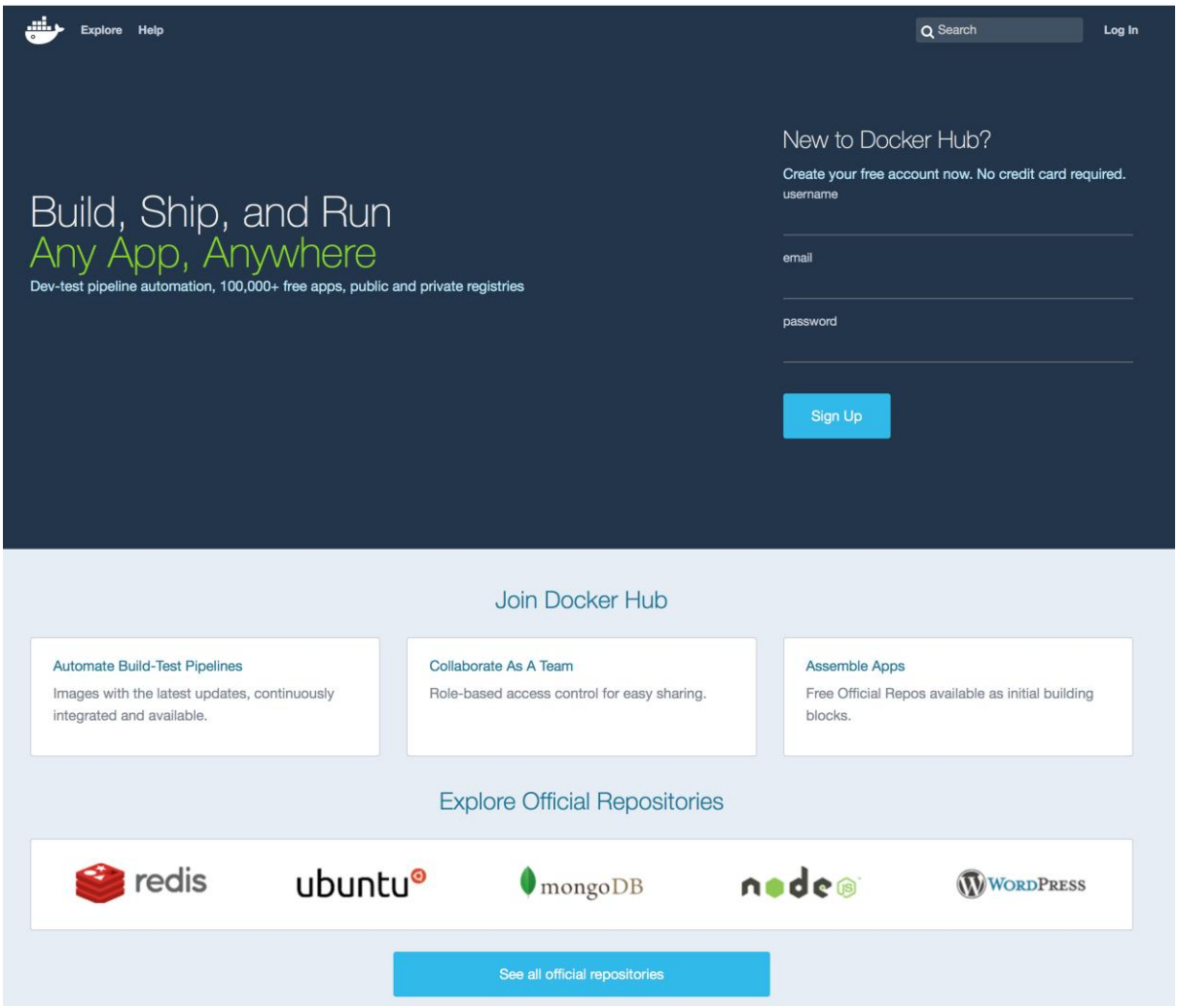
```
==> node-01: INFO          -> Updating module /tmp/modules/stdlib
==> node-01: INFO          -> Updating module /tmp/modules/apt
==> node-01: INFO          -> Updating module /tmp/modules/epel
==> node-01: INFO          -> Updating module /tmp/modules/docker
```

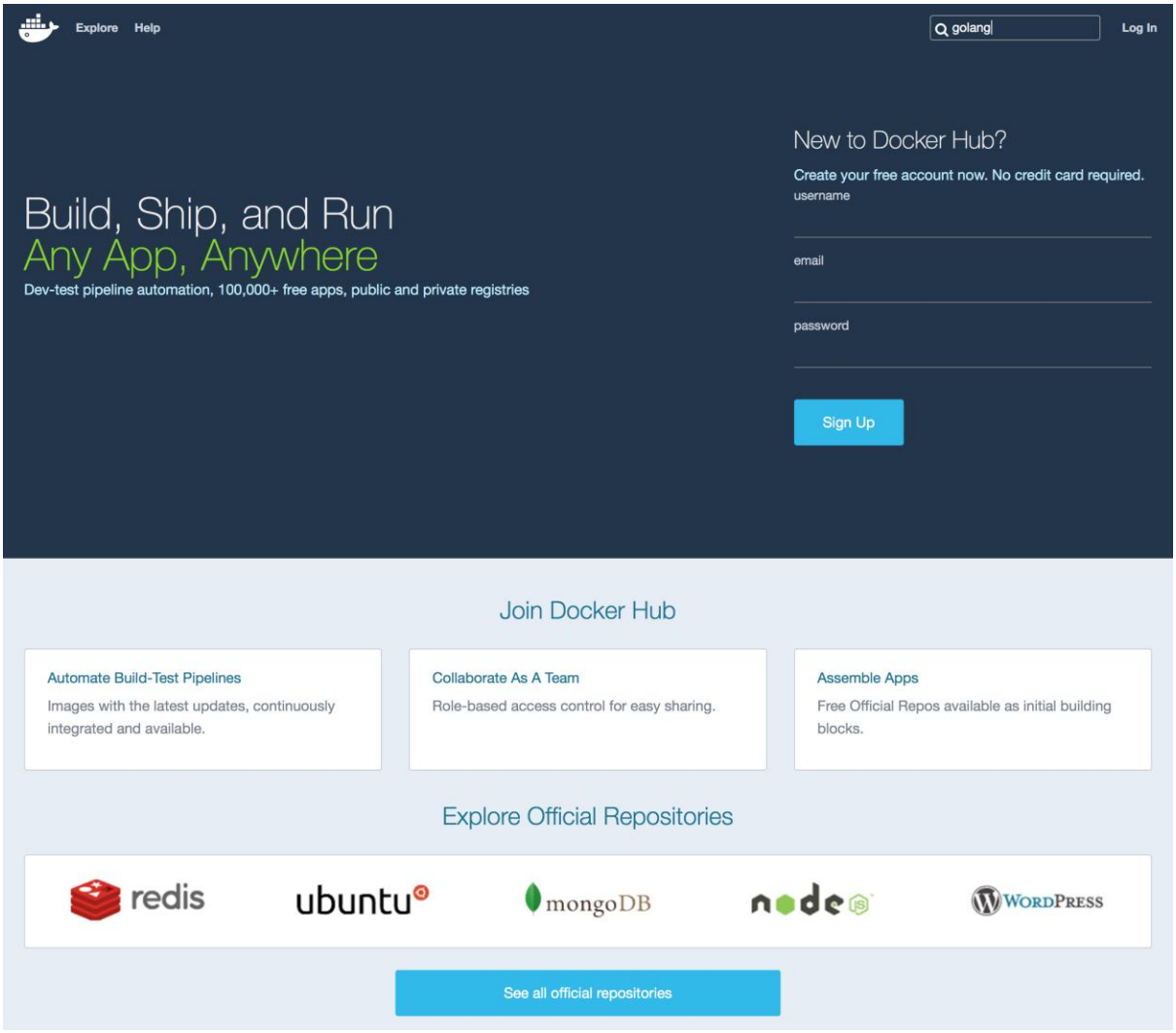
```
✔  [scottcoulton@Scotts-MacBook-Pro] ~/Documents/Local Dev/Vagrant Builds/My book [master|+ 3…1]
14:50 $ vagrant up
```

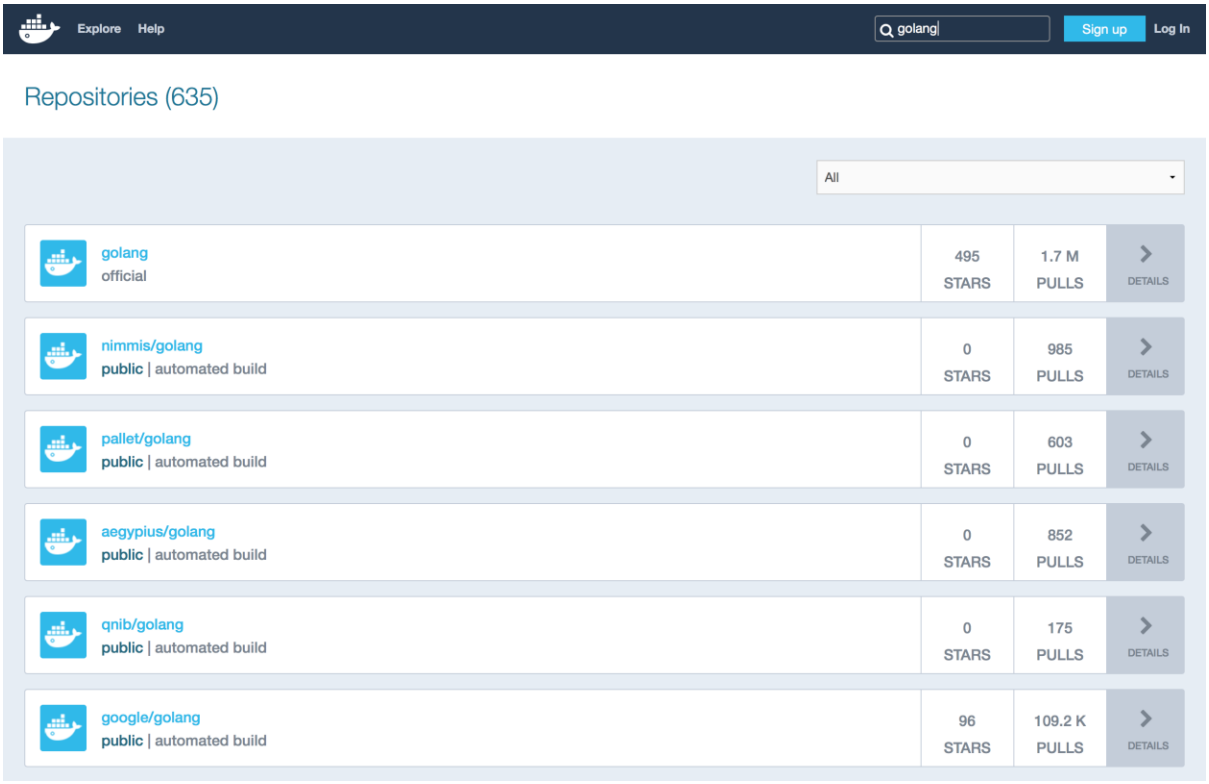
```
==> node-01: Running Puppet with default.pp...
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Notice: Compiled catalog for localhost in environment production in 0.99 seconds
==> node-01: Info: Applying configuration version '1450756659'
==> node-01: Notice: /Stage[main]/Docker::Repos/Yumrepo[docker]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/docker.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Docker::Install/Package[docker]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-01: Notice: /Stage[main]/Docker::Systemd_reload/Exec[docker-systemd-reload]: Triggered 'refresh' from 1 events
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/Service[docker]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Docker::Service/Service[docker]: Unscheduling refresh on Service[docker]
==> node-01: Notice: Finished catalog run in 22.35 seconds
```

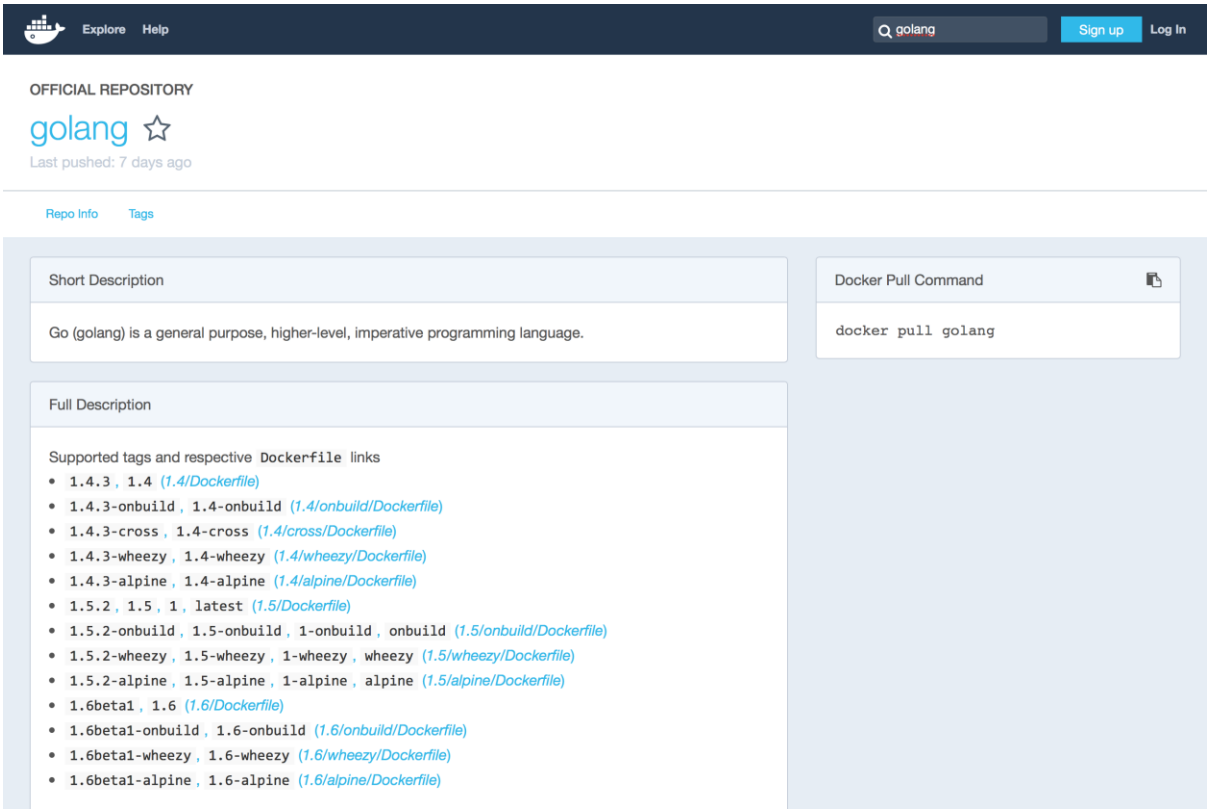
```
[vagrant@node-01 ~]$ sudo -i
[root@node-01 ~]# docker
Usage: docker [OPTIONS] COMMAND [arg...]
       docker daemon [ --help | ... ]
       docker [ --help | -v | --version ]

A self-sufficient runtime for containers.

Options:

  --config=~/.docker                Location of client config files
  -D, --debug=false                 Enable debug mode
  --disable-legacy-registry=false   Do not contact legacy registries
  -H, --host=[]                     Daemon socket(s) to connect to
  -h, --help=false                  Print usage
  -l, --log-level=info              Set the logging level
  --tls=false                       Use TLS; implied by --tlsverify
  --tlscacert=~/.docker/ca.pem      Trust certs signed only by this CA
  --tlscert=~/.docker/cert.pem      Path to TLS certificate file
  --tlskey=~/.docker/key.pem        Path to TLS key file
  --tlsverify=false                 Use TLS and verify the remote
  -v, --version=false               Print version information and quit

Commands:
    attach    Attach to a running container
    build     Build an image from a Dockerfile
    commit    Create a new image from a container's changes
    cp        Copy files/folders between a container and the local filesystem
    create    Create a new container
    diff      Inspect changes on a container's filesystem
    events    Get real time events from the server
    exec      Run a command in a running container
    export    Export a container's filesystem as a tar archive
    history   Show the history of an image
    images    List images
    import    Import the contents from a tarball to create a filesystem image
    info      Display system-wide information
    inspect   Return low-level information on a container or image
    kill      Kill a running container
    load      Load an image from a tar archive or STDIN
    login     Register or log in to a Docker registry
    logout    Log out from a Docker registry
    logs      Fetch the logs of a container
    network   Manage Docker networks
    pause     Pause all processes within a container
    port      List port mappings or a specific mapping for the CONTAINER
    ps        List containers
    pull      Pull an image or a repository from a registry
    push      Push an image or a repository to a registry
    rename    Rename a container
    restart   Restart a container
    rm        Remove one or more containers
    rmi       Remove one or more images
    run       Run a command in a new container
    save      Save an image(s) to a tar archive
    search    Search the Docker Hub for images
    start     Start one or more stopped containers
    stats     Display a live stream of container(s) resource usage statistics
    stop      Stop a running container
    tag       Tag an image into a repository
    top       Display the running processes of a container
    unpause   Unpause all processes within a container
    version   Show the Docker version information
    volume    Manage Docker volumes
    wait      Block until a container stops, then print its exit code

Run 'docker COMMAND --help' for more information on a command.
[root@node-01 ~]#
```

Chapter 2: Working with Docker Hub



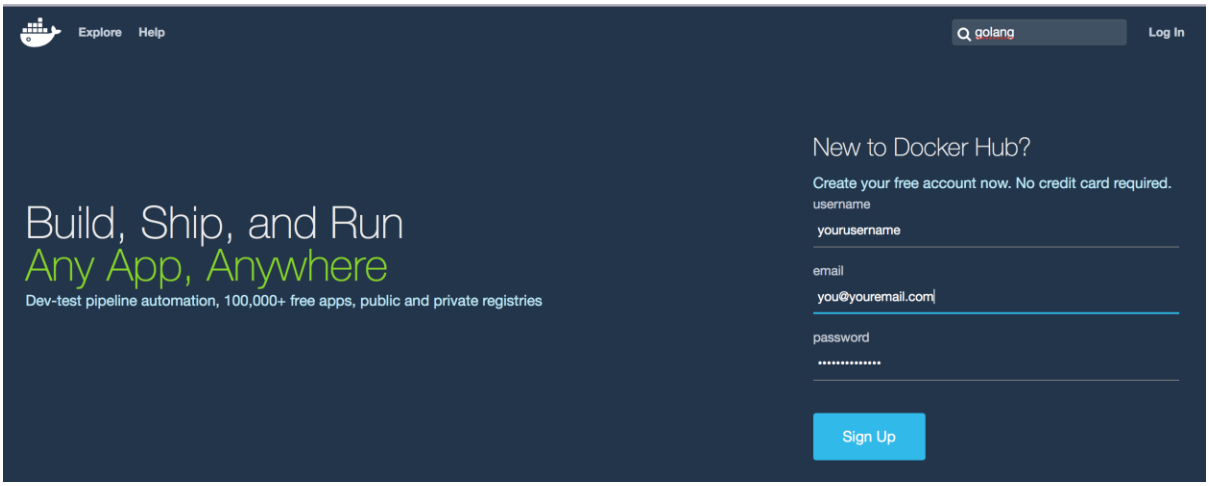

Explore
Help
golang
Log In
Build, Ship, and Run
Any App, Anywhere
Dev-test pipeline automation, 100,000+ free apps, public and private registries
New to Docker Hub?
Create your free account now. No credit card required.
username
email
password
Sign Up
Join Docker Hub
Automate Build-Test Pipelines
Images with the latest updates, continuously integrated and available.
Collaborate As A Team
Role-based access control for easy sharing.
Assemble Apps
Free Official Repos available as initial building blocks.
Explore Official Repositories
redis
ubuntu
mongoDB
node
WordPress
See all official repositories

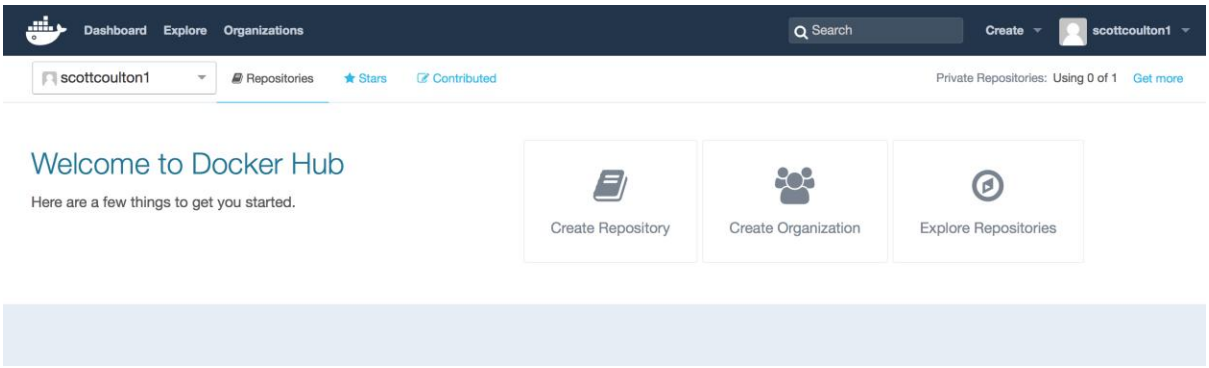

Explore
Help
golang
Sign up
Log In
Repositories (635)
All
golang
official
495
STARS
1.7 M
PULLS
DETAILS
nimmis/golang
public | automated build
0
STARS
985
PULLS
DETAILS
pallet/golang
public | automated build
0
STARS
603
PULLS
DETAILS
aegypius/golang
public | automated build
0
STARS
852
PULLS
DETAILS
qnib/golang
public | automated build
0
STARS
175
PULLS
DETAILS
google/golang
public | automated build
96
STARS
109.2 K
PULLS
DETAILS

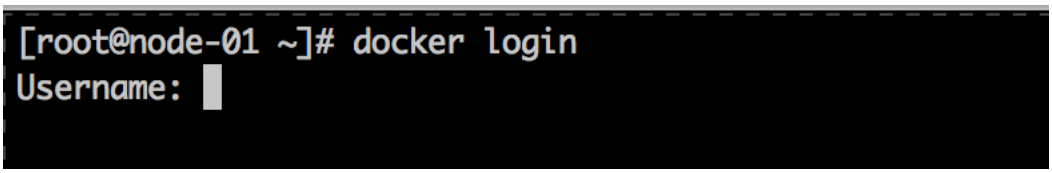


`1.4.3-wheezy`, `1.4-wheezy`

```
$ docker build -t my-golang-app .
```


Explore
Help
golang
Log In
Build, Ship, and Run
Any App, Anywhere
Dev-test pipeline automation, 100,000+ free apps, public and private registries
New to Docker Hub?
Create your free account now. No credit card required.
username
yourusername
email
you@youremail.com
password
Sign Up

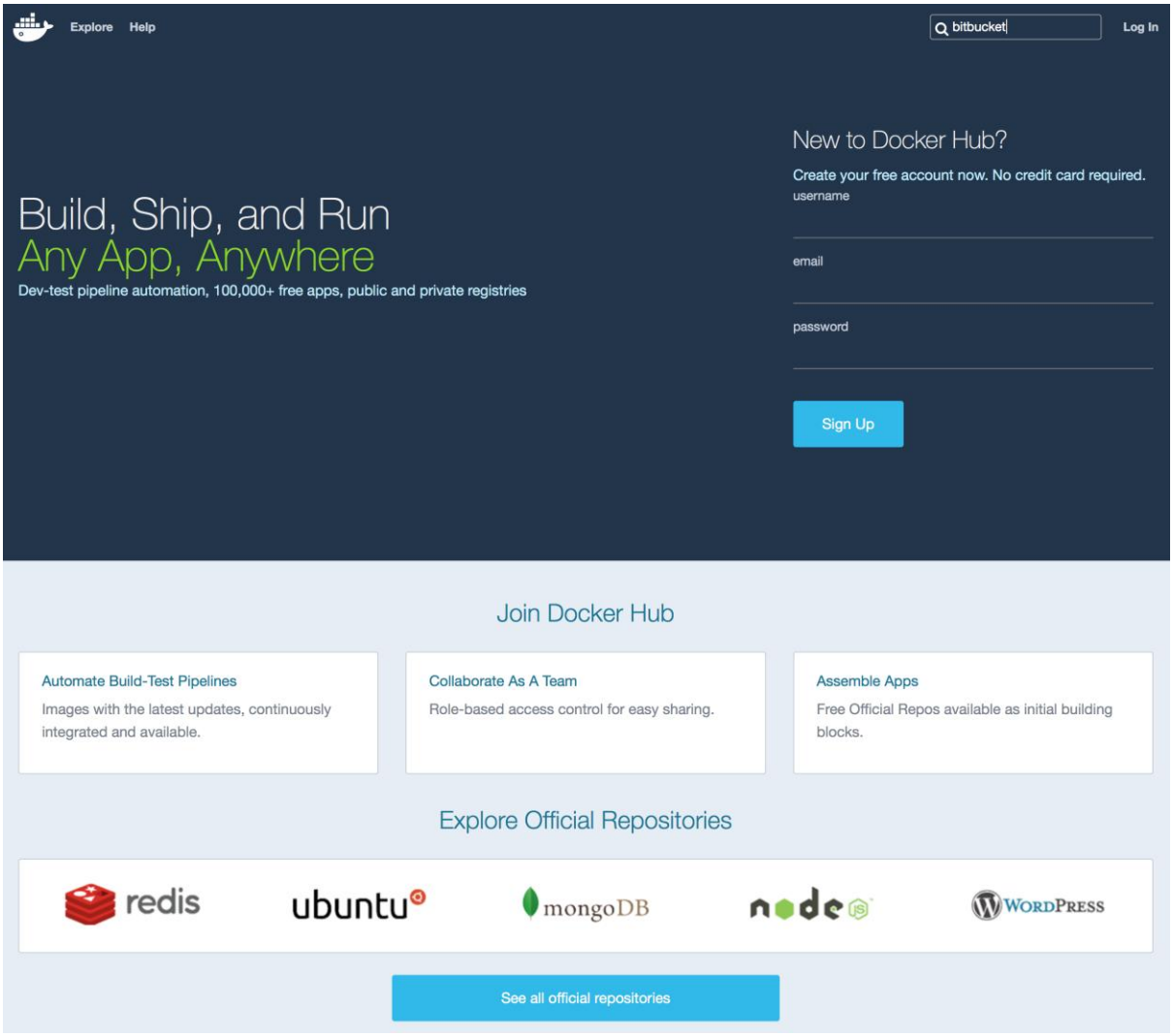

Dashboard
Explore
Organizations
Search
Create
scottcoulton1
scottcoulton1
Repositories
Stars
Contributed
Private Repositories: Using 0 of 1
Get more
Welcome to Docker Hub
Here are a few things to get you started.
Create Repository
Create Organization
Explore Repositories

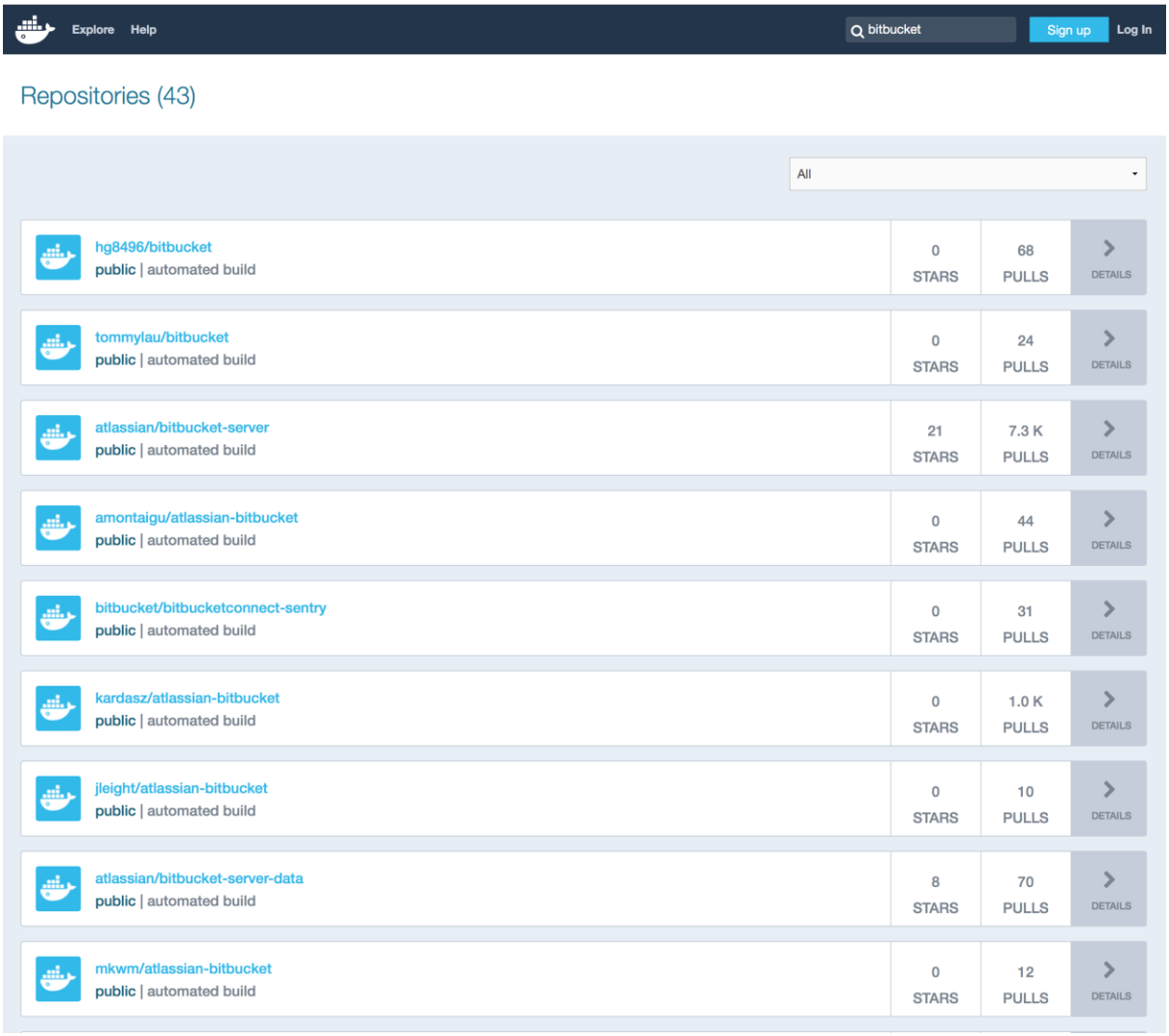

[root@node-01 ~]# docker login
Username:

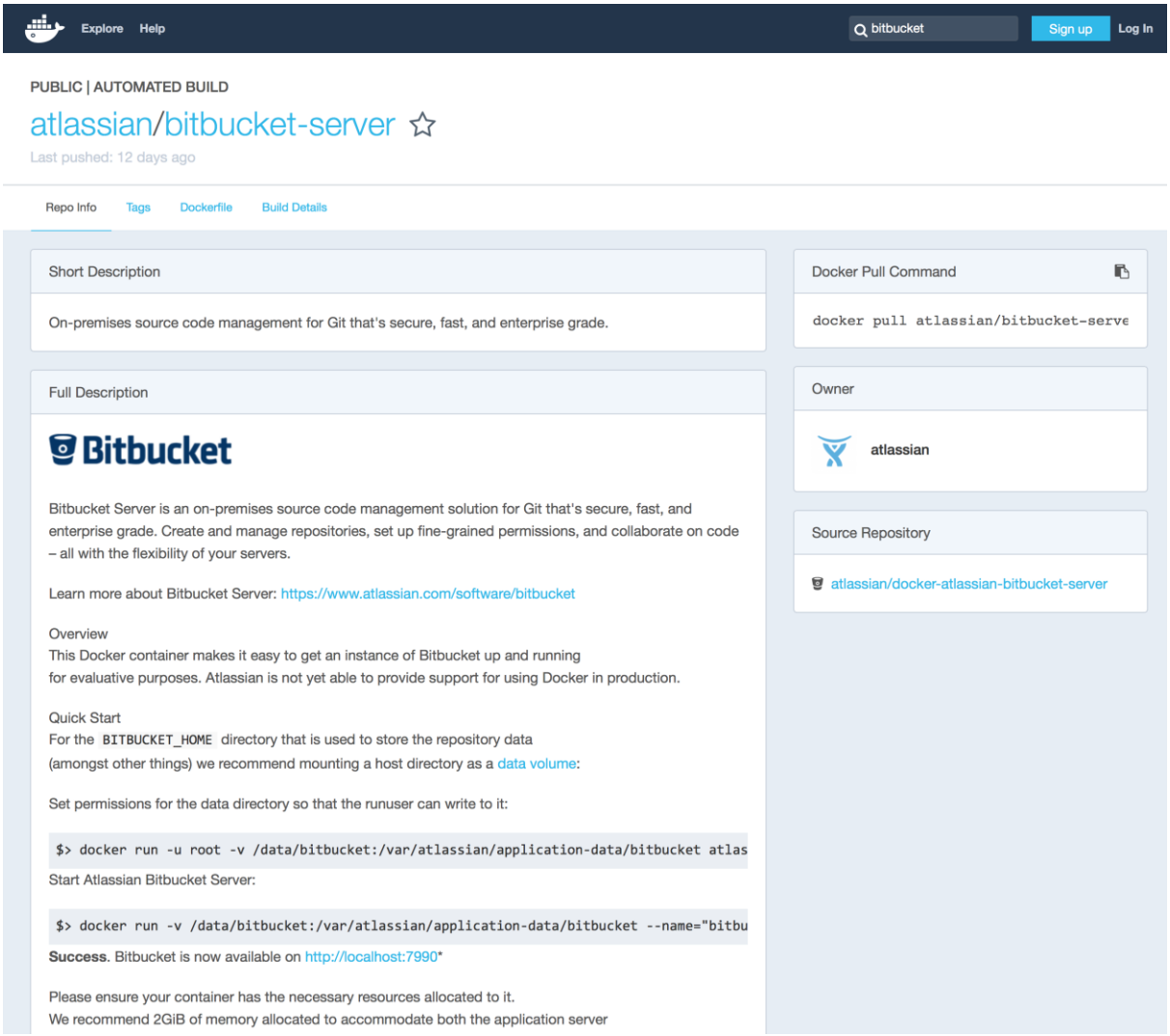
```
[root@node-01 ~]# docker login
Username: scottyc
Password:
```

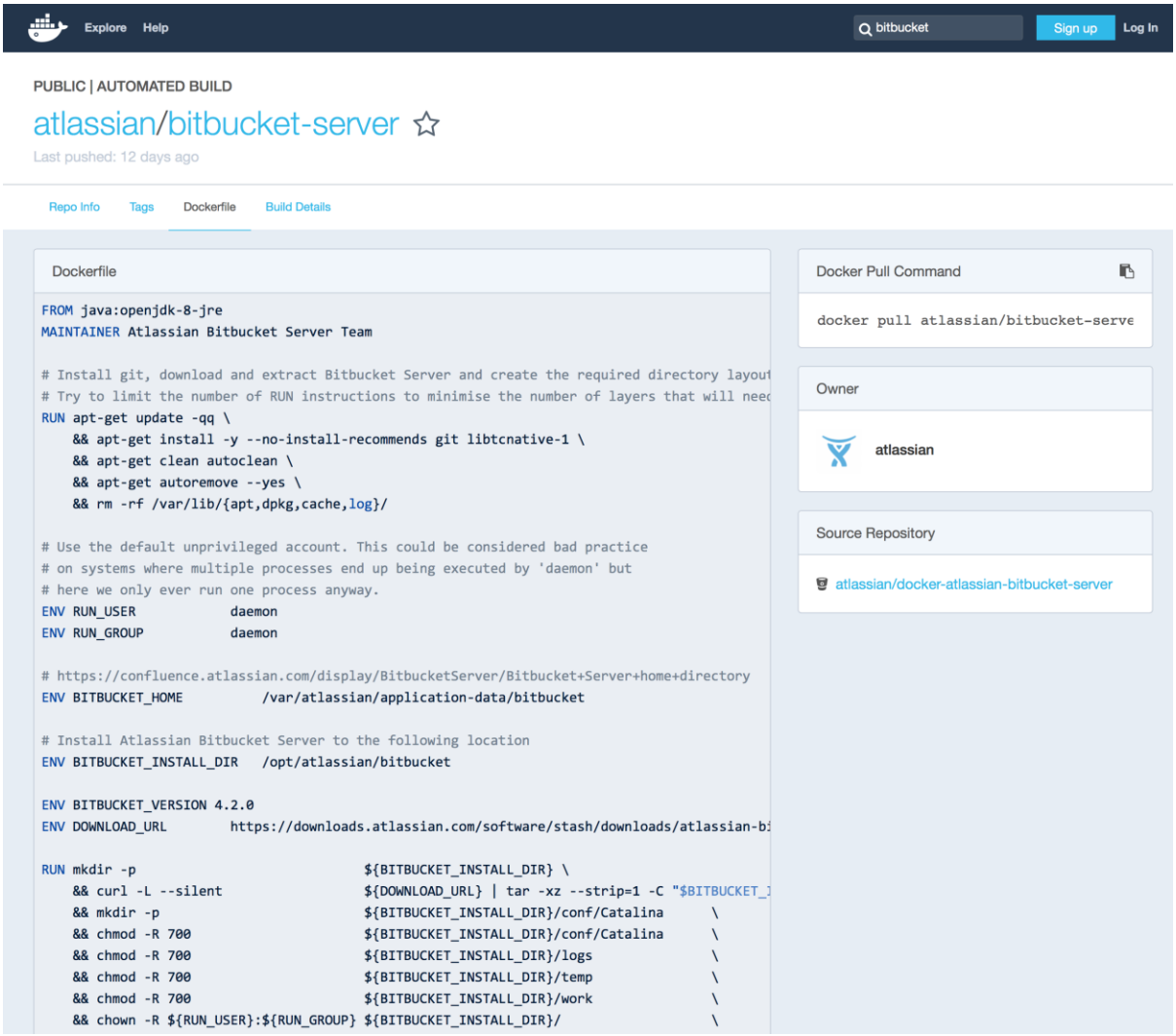
```
[root@node-01 ~]# docker login
Username: scottyc
Password:
Email:
```

```
[root@node-01 ~]# docker login
Username: scottyc
Password:
Email: scott.coulton@gmail.com
WARNING: login credentials saved in /root/.docker/config.json
Login Succeeded
[root@node-01 ~]#
```

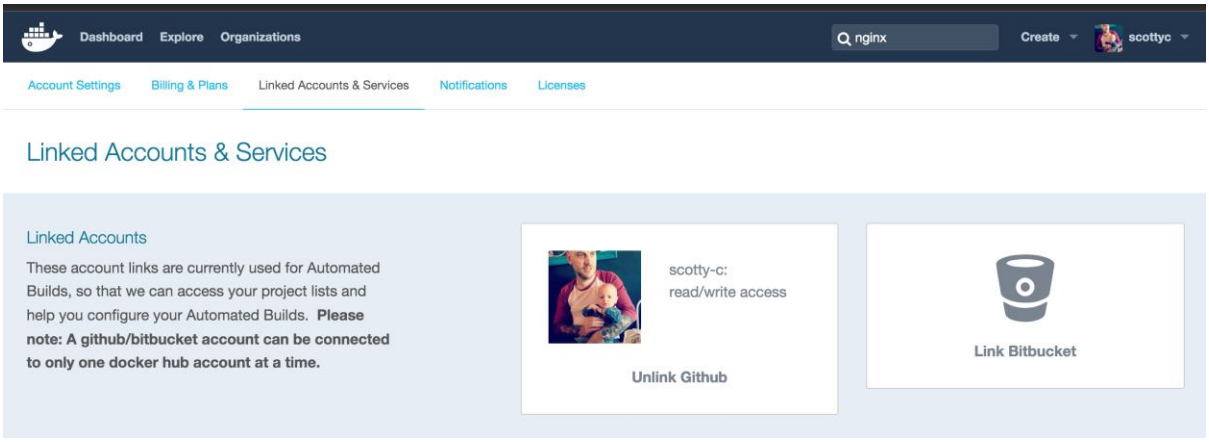

Explore
Help
bitbucket
Log In
Build, Ship, and Run
Any App, Anywhere
Dev-test pipeline automation, 100,000+ free apps, public and private registries
New to Docker Hub?
Create your free account now. No credit card required.
username
email
password
Sign Up
Join Docker Hub
Automate Build-Test Pipelines
Images with the latest updates, continuously integrated and available.
Collaborate As A Team
Role-based access control for easy sharing.
Assemble Apps
Free Official Repos available as initial building blocks.
Explore Official Repositories
redis
ubuntu
mongoDB
node
WordPress
See all official repositories

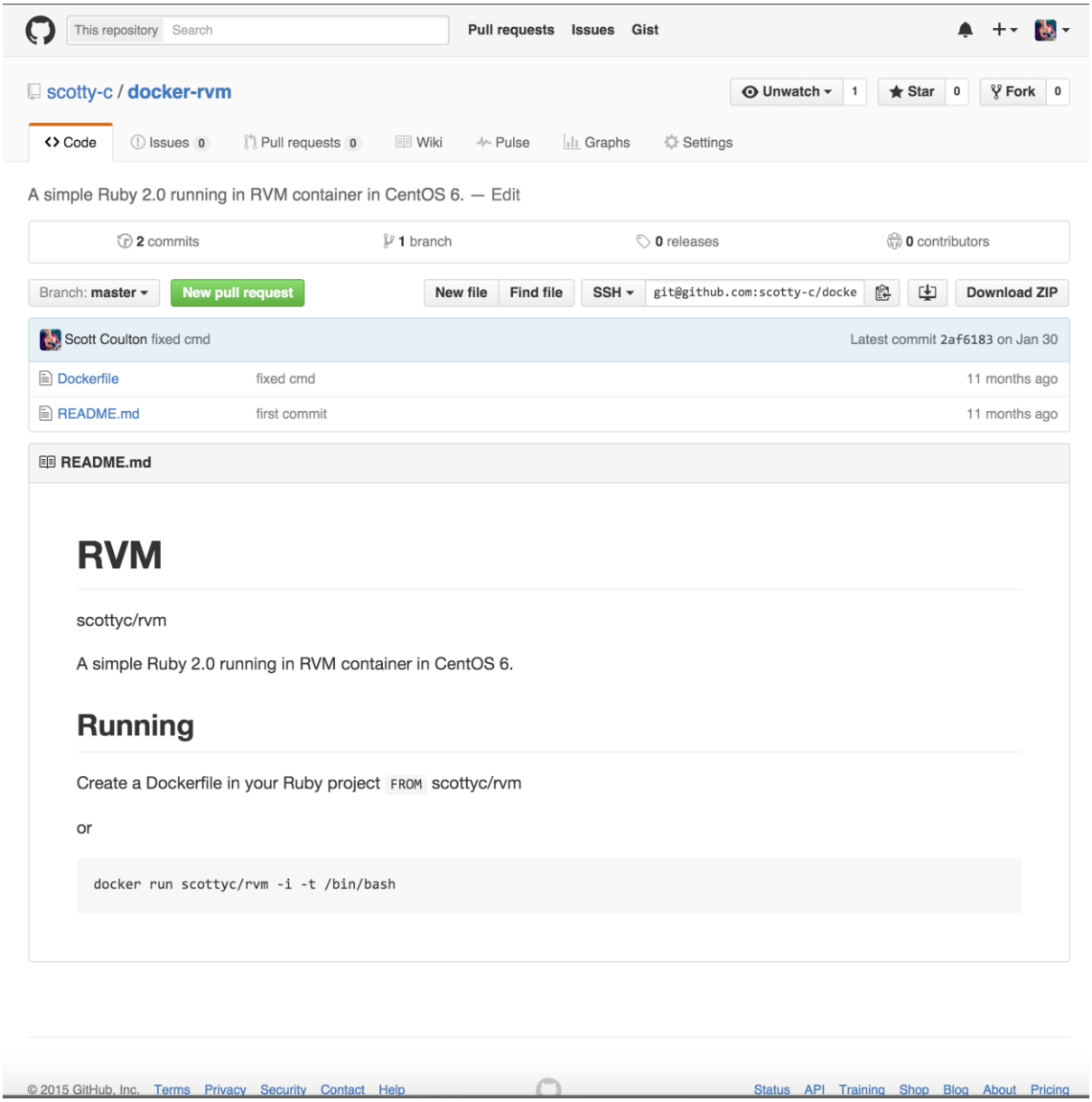

Explore
Help
bitbucket
Sign up
Log In
Repositories (43)
All
hg8496/bitbucket
public | automated build
0 STARS
68 PULLS
DETAILS
tommylau/bitbucket
public | automated build
0 STARS
24 PULLS
DETAILS
atlassian/bitbucket-server
public | automated build
21 STARS
7.3 K PULLS
DETAILS
amontaigu/atlassian-bitbucket
public | automated build
0 STARS
44 PULLS
DETAILS
bitbucket/bitbucketconnect-sentry
public | automated build
0 STARS
31 PULLS
DETAILS
kardasz/atlassian-bitbucket
public | automated build
0 STARS
1.0 K PULLS
DETAILS
jleight/atlassian-bitbucket
public | automated build
0 STARS
10 PULLS
DETAILS
atlassian/bitbucket-server-data
public | automated build
8 STARS
70 PULLS
DETAILS
mkwm/atlassian-bitbucket
public | automated build
0 STARS
12 PULLS
DETAILS

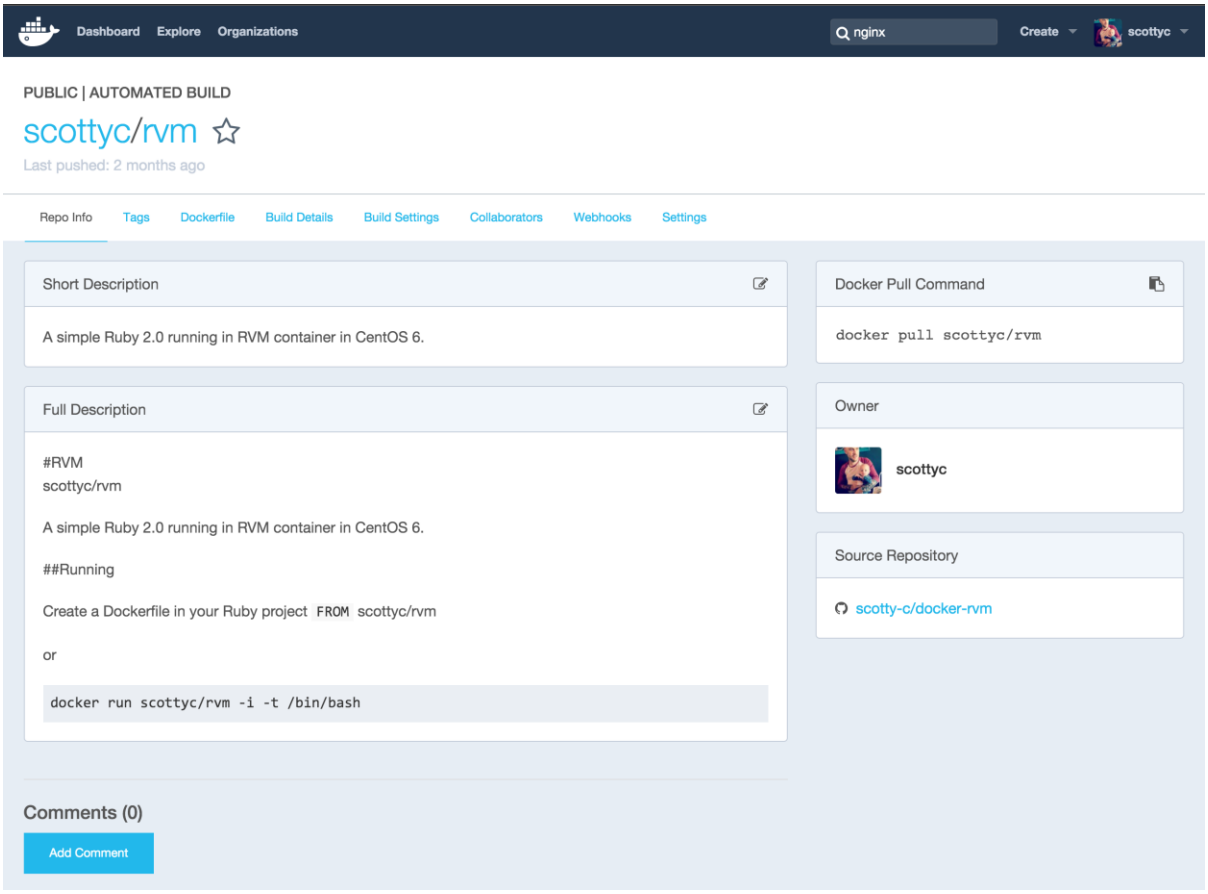

Explore
Help
bitbucket
Sign up
Log In
PUBLIC | AUTOMATED BUILD
atlassian/bitbucket-server
Last pushed: 12 days ago
Repo Info
Tags
Dockerfile
Build Details
Short Description
On-premises source code management for Git that's secure, fast, and enterprise grade.
Full Description
Bitbucket
Bitbucket Server is an on-premises source code management solution for Git that's secure, fast, and enterprise grade. Create and manage repositories, set up fine-grained permissions, and collaborate on code – all with the flexibility of your servers.
Learn more about Bitbucket Server: https://www.atlassian.com/software/bitbucket
Overview
This Docker container makes it easy to get an instance of Bitbucket up and running
for evaluative purposes. Atlassian is not yet able to provide support for using Docker in production.
Quick Start
For the BITBUCKET_HOME directory that is used to store the repository data
(amongst other things) we recommend mounting a host directory as a data volume:
Set permissions for the data directory so that the runuser can write to it:
$> docker run -u root -v /data/bitbucket:/var/atlassian/application-data/bitbucket atlas
Start Atlassian Bitbucket Server:
$> docker run -v /data/bitbucket:/var/atlassian/application-data/bitbucket --name="bitbu
Success. Bitbucket is now available on http://localhost:7990*
Please ensure your container has the necessary resources allocated to it.
We recommend 2GiB of memory allocated to accommodate both the application server
Docker Pull Command
docker pull atlassian/bitbucket-serve
Owner
atlassian
Source Repository
atlassian/docker-atlassian-bitbucket-server

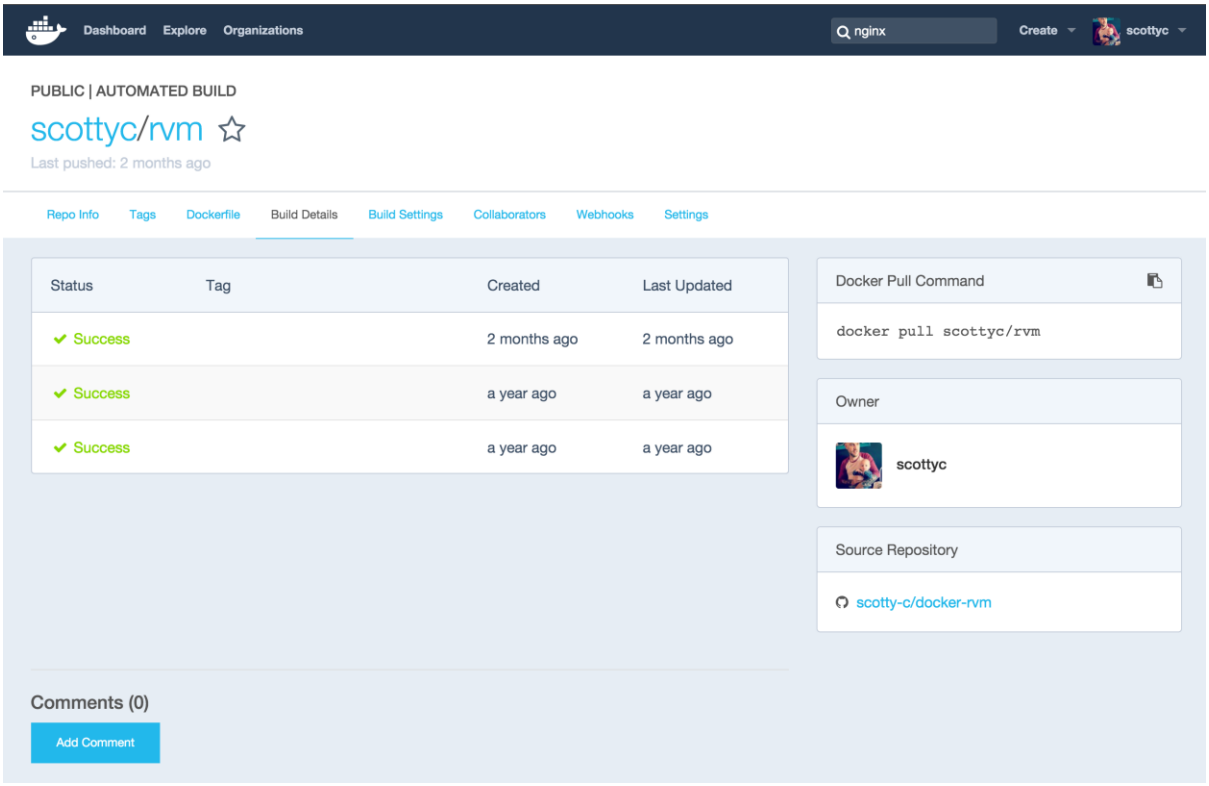

Explore
Help
bitbucket
Sign up
Log In
PUBLIC | AUTOMATED BUILD
atlassian/bitbucket-server
Last pushed: 12 days ago
Repo Info
Tags
Dockerfile
Build Details
Dockerfile
FROM java:openjdk-8-jre
MAINTAINER Atlassian Bitbucket Server Team
Install git, download and extract Bitbucket Server and create the required directory layout
Try to limit the number of RUN instructions to minimise the number of layers that will need
RUN apt-get update -qq \
&& apt-get install -y --no-install-recommends git libtcnative-1 \
&& apt-get clean autoclean \
&& apt-get autoremove --yes \
&& rm -rf /var/lib/{apt,dpkg,cache,log}/
# Use the default unprivileged account. This could be considered bad practice
# on systems where multiple processes end up being executed by 'daemon' but
# here we only ever run one process anyway.
ENV RUN_USER daemon
ENV RUN_GROUP daemon
https://confluence.atlassian.com/display/BitbucketServer/Bitbucket+Server+home+directory
ENV BITBUCKET_HOME /var/atlassian/application-data/bitbucket
Install Atlassian Bitbucket Server to the following location
ENV BITBUCKET_INSTALL_DIR /opt/atlassian/bitbucket
ENV BITBUCKET_VERSION 4.2.0
ENV DOWNLOAD_URL https://downloads.atlassian.com/software/stash/downloads/atlassian-b
RUN mkdir -p ${BITBUCKET_INSTALL_DIR} \
&& curl -L --silent ${DOWNLOAD_URL} | tar -xz --strip=1 -C "$BITBUCKET_
&& mkdir -p ${BITBUCKET_INSTALL_DIR}/conf/Catalina \
&& chmod -R 700 ${BITBUCKET_INSTALL_DIR}/conf/Catalina \
&& chmod -R 700 ${BITBUCKET_INSTALL_DIR}/logs \
&& chmod -R 700 ${BITBUCKET_INSTALL_DIR}/temp \
&& chmod -R 700 ${BITBUCKET_INSTALL_DIR}/work \
&& chown -R ${RUN_USER}:${RUN_GROUP} ${BITBUCKET_INSTALL_DIR}/ \
Docker Pull Command
docker pull atlassian/bitbucket-serve
Owner
atlassian
Source Repository
atlassian/docker-atlassian-bitbucket-server

```
[root@node-01 ~]# docker search bitbucket
NAME                                      DESCRIPTION                                     STARS     OFFICIAL   AUTOMATED
atlassian/bitbucket-server                On-premises source code management for Git...   21                   [OK]
atlassian/bitbucket-server-data           Data volume container for Bitbucket Server      8                    [OK]
dweomer/atlassian-bitbucket               Atlassian Bitbucker Server, Dockerized!         1                    [OK]
nkatsaros/atlassian-bitbucket             Code, Manage, Collaborate                       1                    [OK]
kazssym/bitbucket-webhook-example         Example web application of the Bitbucket W...   1
jleight/atlassian-bitbucket               Container for Atlassian Bitbucket® Server.      0                    [OK]
mkwm/atlassian-bitbucket                  Atlassian Bitbucket                             0                    [OK]
hillrunner2008/docker-jenkins-bitbucket   docker-jenkins-bitbucket                        0                    [OK]
premiumminds/bitbucket-backup-tool        Bitbucket backup and restore tool               0                    [OK]
lukaspronto/bitbucket-slack-pr-hook       BitBucket Pull Request notification hook f...   0                    [OK]
inanimate/bitbucket-backup-client         A containerized bitbucket backup client ma...   0                    [OK]
dunkelfrosch/bitbucket                    This repository provide the latest version...   0
seibertmedia/atlassian-bitbucket          Atlassian Bitbucket                             0
naeemattari7/test-docker-bitbucket                                                        0                    [OK]
markwigg/nginx-test-bitbucket             test app for a basic nginx site                 0                    [OK]
ynoami/bitbucket                                                                          0
yohanliyanage/bitbucket-hookfilter        Jenkins Bitbucket Hook Filter                   0
inventame/base-bitbucket                                                                  0
surecomms/alpine-bitbucket                                                                0
bitbucket/hipbucket_base                                                                  0
hg8496/bitbucket                          Bitbucket Server                                0                    [OK]
tommylau/bitbucket                        Bitbucket Server                                0                    [OK]
amontaigu/atlassian-bitbucket             https://github.com/AlbanMontaigu/docker-at...   0                    [OK]
bitbucket/bitbucketconnect-sentry                                                         0                    [OK]
kardasz/atlassian-bitbucket               Atlassian Bitbucket                             0                    [OK]
```




This repository Search
Pull requests Issues Gist
scotty-c / docker-rvm
Unwatch 1 Star 0 Fork 0
Code Issues 0 Pull requests 0 Wiki Pulse Graphs Settings
A simple Ruby 2.0 running in RVM container in CentOS 6. — Edit
2 commits 1 branch 0 releases 0 contributors
Branch: master New pull request
New file Find file SSH git@github.com:scotty-c/docke Download ZIP
Scott Coulton fixed cmd
Latest commit 2af6183 on Jan 30
Dockerfile fixed cmd 11 months ago
README.md first commit 11 months ago
README.md
RVM
scottyc/rvm
A simple Ruby 2.0 running in RVM container in CentOS 6.
Running
Create a Dockerfile in your Ruby project FROM scottyc/rvm
or
docker run scottyc/rvm -i -t /bin/bash
© 2015 GitHub, Inc. Terms Privacy Security Contact Help
Status API Training Shop Blog About Pricing

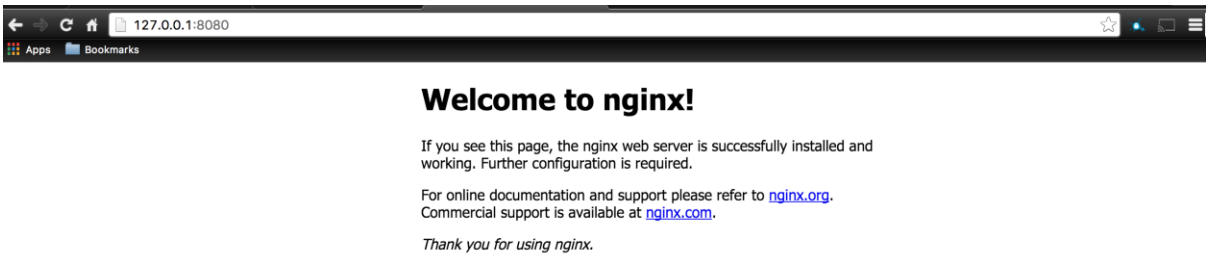

Dashboard
Explore
Organizations
nginx
Create
scottyc
PUBLIC | AUTOMATED BUILD
scottyc/rvm
Last pushed: 2 months ago
Repo Info
Tags
Dockerfile
Build Details
Build Settings
Collaborators
Webhooks
Settings
Short Description
A simple Ruby 2.0 running in RVM container in CentOS 6.
Full Description
#RVM
scottyc/rvm
A simple Ruby 2.0 running in RVM container in CentOS 6.
##Running
Create a Dockerfile in your Ruby project FROM scottyc/rvm
or
docker run scottyc/rvm -i -t /bin/bash
Docker Pull Command
docker pull scottyc/rvm
Owner
scottyc
Source Repository
scotty-c/docker-rvm
Comments (0)
Add Comment




```
1  FROM nginx
2
3  MAINTAINER Scott Coulton
```

```
1  FROM nginx
2
3  MAINTAINER Scott Coulton
4
5  RUN apt-get -qqy update
```

```
[root@node-01 vagrant]# docker build -t scottyc/nginx .
Sending build context to Docker daemon 98.82 kB
Step 1 : FROM nginx
latest: Pulling from library/nginx

9ee13ca3b908: Pull complete
23cb15b0fcec: Pull complete
62df5e17dafa: Pull complete
d65968c1aa44: Pull complete
f5bb1dddc876: Pull complete
1526247f349d: Pull complete
2e518e3d3fad: Pull complete
0e07123e6531: Pull complete
21656a3c1256: Pull complete
f608475c6c65: Pull complete
1b6c0a20b353: Pull complete
5328fdfe9b8e: Pull complete
Digest: sha256:a79db4b83c0dbad9542d5442002ea294aa77014a3dfa67160d8a55874a5520cc
Status: Downloaded newer image for nginx:latest
 ---> 5328fdfe9b8e
Step 2 : MAINTAINER Scott Coulton
 ---> Running in 5c30c184f0d6
 ---> d2843a2a5a53
Removing intermediate container 5c30c184f0d6
Step 3 : RUN apt-get -qqy update
 ---> Running in b8fd1c675494
 ---> 821117a98fcd
Removing intermediate container b8fd1c675494
Successfully built 821117a98fcd
```



```
#!/usr/bin/ruby env

require "socket"
$hostname = Socket.gethostname

forge 'http://forge.puppetlabs.com'


mod 'puppetlabs/stdlib', '4.1.0'
mod 'puppetlabs/apt', '2.2.1'
mod 'stahnma/epel'
mod 'garethr/docker', :git => "https://github.com/scotty-c/garethr-docker.git"
mod 'stankevich/python'
```

```
[vagrant@node-01 ~]$ sudo -i
[root@node-01 ~]# docker-compose
Define and run multi-container applications with Docker.

Usage:
  docker-compose [-f=<arg>...] [options] [COMMAND] [ARGS...]
  docker-compose -h|--help

Options:
  -f, --file FILE           Specify an alternate compose file (default: docker-compose.yml)
  -p, --project-name NAME   Specify an alternate project name (default: directory name)
  --x-networking            (EXPERIMENTAL) Use new Docker networking functionality.
                            Requires Docker 1.9 or later.
  --x-network-driver DRIVER (EXPERIMENTAL) Specify a network driver (default: "bridge").
                            Requires Docker 1.9 or later.
  --verbose                 Show more output
  -v, --version             Print version and exit

Commands:
  build              Build or rebuild services
  help               Get help on a command
  kill               Kill containers
  logs               View output from containers
  pause              Pause services
  port               Print the public port for a port binding
  ps                 List containers
  pull               Pulls service images
  restart            Restart services
  rm                 Remove stopped containers
  run                Run a one-off command
  scale              Set number of containers for a service
  start              Start services
  stop               Stop services
  unpause            Unpause services
  up                 Create and start containers
  migrate-to-labels  Recreate containers to add labels
  version            Show the Docker-Compose version information
```

```
1   nginx:
2       image: nginx
3       hostname: nginx
4       ports:
5         - "80:80"
```

```
[root@node-01 vagrant]# docker-compose up -d
Pulling nginx (nginx:latest)...
latest: Pulling from library/nginx
9ee13ca3b908: Pull complete
23cb15b0fcec: Pull complete
62df5e17dafa: Pull complete
d65968c1aa44: Pull complete
f5bb1dddc876: Pull complete
1526247f349d: Pull complete
2e518e3d3fad: Pull complete
0e07123e6531: Pull complete
21656a3c1256: Pull complete
f608475c6c65: Pull complete
1b6c0a20b353: Pull complete
5328fdfe9b8e: Pull complete
Digest: sha256:a79db4b83c0dbad9542d5442002ea294aa77014a3dfa67160d8a55874a5520cc
Status: Downloaded newer image for nginx:latest
Creating vagrant_nginx_1
```

Welcome to nginx!

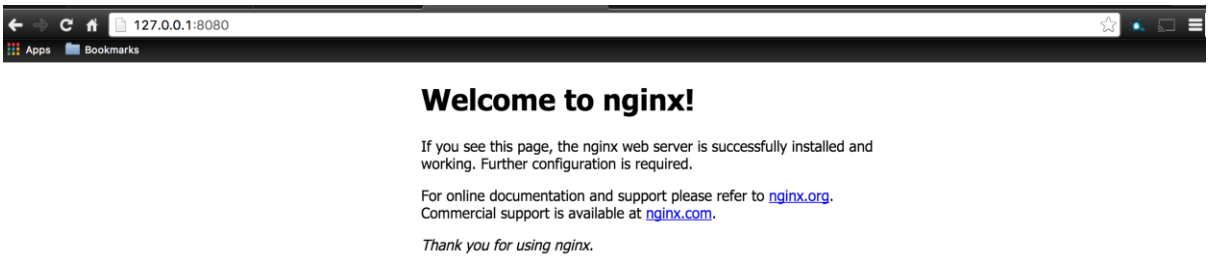
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

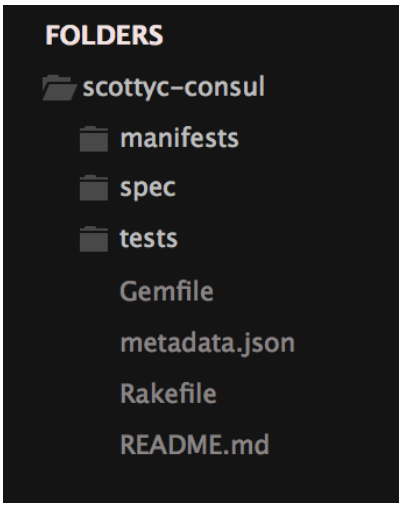
```
node 'node-01' {

  include docker

  docker::run { 'nginx':
  image    => 'nginx',
  ports    => ['80', '80'],
  hostname => 'nginx',
  }
}
```

```
==> node-01: Running provisioner: puppet...
==> node-01: Running Puppet with default.pp...
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Notice: Compiled catalog for localhost in environment production in 1.28 seconds
==> node-01: Info: Applying configuration version '1451523602'
==> node-01: Notice: /Stage[main]/Main/Node[node-01]/Docker::Run[nginx]/File[/etc/systemd/system/docker-nginx.service]/ensure: created
==> node-01: Info: /Stage[main]/Main/Node[node-01]/Docker::Run[nginx]/File[/etc/systemd/system/docker-nginx.service]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-01: Info: /Stage[main]/Main/Node[node-01]/Docker::Run[nginx]/File[/etc/systemd/system/docker-nginx.service]: Scheduling refresh of Service[docker-nginx]
==> node-01: Notice: /Stage[main]/Docker::Systemd_reload/Exec[docker-systemd-reload]: Triggered 'refresh' from 1 events
==> node-01: Notice: /Stage[main]/Main/Node[node-01]/Docker::Run[nginx]/Service[docker-nginx]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Main/Node[node-01]/Docker::Run[nginx]/Service[docker-nginx]: Unscheduling refresh on Service[docker-nginx]
==> node-01: Notice: Finished catalog run in 1.14 seconds
```



Chapter 3 : Building a Single Container Application

```
[root@node-01 ~]# puppet module generate scottyc-consul
We need to create a metadata.json file for this module.  Please answer the
following questions; if the question is not applicable to this module, feel free
to leave it blank.

Puppet uses Semantic Versioning (semver.org) to version modules.
What version is this module?  [0.1.0]
-->

Who wrote this module?  [scottyc]
-->

What license does this module code fall under?  [Apache 2.0]
-->

How would you describe this module in a single sentence?
--> This is a module the runs Conusl in a Docker container

Where is this module's source code repository?
-->

Where can others go to learn more about this module?
-->

Where can others go to file issues about this module?
-->

----------------------------------------
{
  "name": "scottyc-consul",
  "version": "0.1.0",
  "author": "scottyc",
  "summary": "This is a module the runs Conusl in a Docker container",
  "license": "Apache 2.0",
  "source": "",
  "project_page": null,
  "issues_url": null,
  "dependencies": [
    {"name":"puppetlabs-stdlib","version_requirement":">= 1.0.0"}
  ]
}
----------------------------------------

About to generate this metadata; continue? [n/Y]
--> y

Notice: Generating module at /root/scottyc-consul...
Notice: Populating templates...
Finished; module generated in scottyc-consul.
scottyc-consul/Gemfile
scottyc-consul/Rakefile
scottyc-consul/manifests
scottyc-consul/manifests/init.pp
scottyc-consul/spec
scottyc-consul/spec/classes
scottyc-consul/spec/classes/init_spec.rb
scottyc-consul/spec/spec_helper.rb
scottyc-consul/tests
scottyc-consul/tests/init.pp
scottyc-consul/README.md
scottyc-consul/metadata.json
[root@node-01 ~]#
```



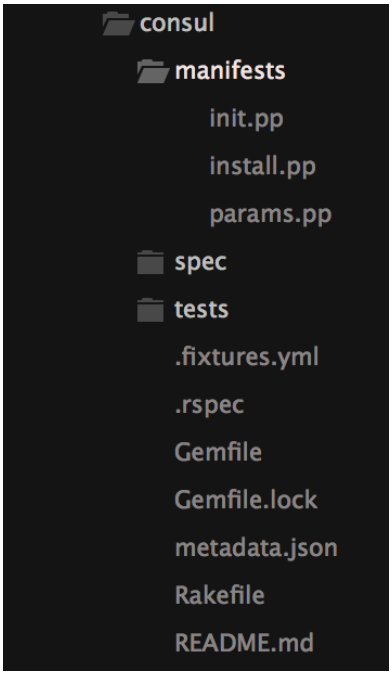

```
fixtures:
    symlinks:
        consul: "#{source_dir}"
    forge_modules:
        vcsrepo: puppetlabs/vcsrepo
        stdlib: puppetlabs/stdlib
        golang: scottyc/golang
        python: stankevich/python
    repositories:
        docker: https://github.com/scotty-c/garethr-docker.git

```

```
--require spec_helper
--pattern spec/*/*_spec.rb

```

```
[root@node-01 scottyc-consul]# bundle install
Don't run Bundler as root. Bundler can ask for sudo if it is needed, and installing your bundle as root will break this application for all non-root users on this machine.
Fetching gem metadata from https://rubygems.org/.........
Fetching version metadata from https://rubygems.org/..
Resolving dependencies...
Rubygems 2.0.14 is not threadsafe, so your gems will be installed one at a time. Upgrade to Rubygems 2.1.0 or higher to enable parallel gem installation.
Installing rake 10.5.0
Installing CFPropertyList 2.2.8
Installing diff-lcs 1.2.5
Installing facter 2.4.4
Installing json_pure 1.8.3
Installing metaclass 0.0.4
Installing puppet-lint 1.1.0
Installing rspec-support 3.4.1
Using bundler 1.11.2
Installing puppet-syntax 2.0.0
Installing hiera 3.0.5
Installing mocha 1.1.0
Installing rspec-core 3.4.1
Installing rspec-expectations 3.4.0
Installing rspec-mocks 3.4.1
Installing puppet 4.3.1
Installing rspec 3.4.0
Installing rspec-puppet 2.3.0
Installing puppetlabs_spec_helper 1.0.1
Bundle complete! 4 Gemfile dependencies, 19 gems now installed.
Use `bundle show [gemname]` to see where a bundled gem is installed.
```

```ruby
require 'spec_helper'
describe 'consul' do

  let(:facts) { {:osfamily => 'RedHat', :operatingsystemrelease => 'RedHat Linux release 7.0'}}

  context 'with defaults for all parameters' do
    it { should contain_class('consul') }
  end
end

```

```json
{
  "name": "scottyc-consul",
  "version": "0.1.0",
  "author": "scottyc",
  "summary": "This is a module the runs Conusl in a Docker container",
  "license": "Apache 2.0",
  "source": "",
  "project_page": null,
  "issues_url": null,
  "dependencies": [
    { "name": "garethr/docker", "version_requirement": ">= 4.0.0" }
  ]
}


```




```
1  class consul::params {
2
3    $docker_version          = '1.9.1-1.el7.centos'
4    $docker_tcp_bind         = 'tcp://127.0.0.1:4243'
5    $docker_image            = 'scottyc/consul'
6    $container_hostname      = 'consul'
7    $consul_advertise        = $::ipaddress_enp0s8
8    $consul_bootstrap_expect = '1'
9  }
10
```

```
class consul::install {

  package { 'device-mapper-libs':
    ensure => installed,
    }

  class { 'docker':
    version     => $consul::docker_version,
    tcp_bind    => $consul::docker_tcp_bind,
    socket_bind => 'unix:///var/run/docker.sock',
    require     => Package['device-mapper-libs']
    } ->

  docker::image { $consul::docker_image : } ->

  docker::run { $consul::container_hostname:
    image    => $consul::docker_image,
    hostname => $consul::container_hostname,
    command  => "-server --advertise ${consul::consul_advertise} -bootstrap-expect ${consul::consul_bootstrap_expect}",
    ports    => ['8301:8301', '8301:8301/udp', '8302:8302', '8302:8302/udp', '8400:8400', '8500:8500', '53:53/udp']
    }
}
```

```
package { 'device-mapper-libs':
  ensure => installed,
  }

class { 'docker':
  version     => $consul::docker_version,
  tcp_bind    => $consul::docker_tcp_bind,
  socket_bind => 'unix:///var/run/docker.sock',
  require     => Package['device-mapper-libs']
  } ->
```

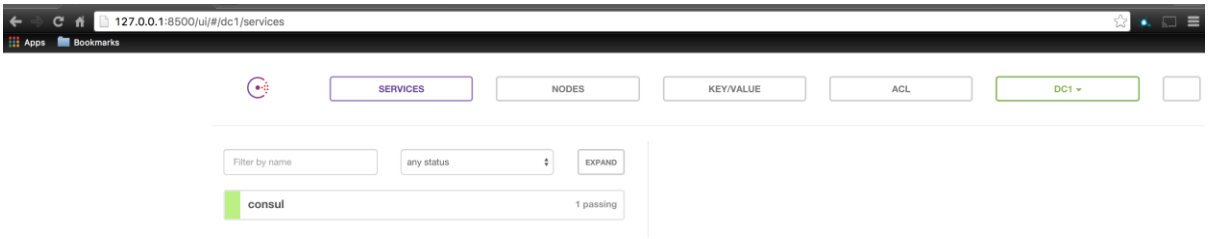
```
  docker::image { $consul::docker_image : } ->

  docker::run { $consul::container_hostname:
    image    => $consul::docker_image,
    hostname => $consul::container_hostname,
    command  => "-server --advertise ${consul::consul_advertise} -bootstrap-expect ${consul::consul_bootstrap_expect}",
    ports    => ['8301:8301', '8301:8301/udp', '8302:8302', '8302:8302/udp', '8400:8400', '8500:8500', '53:53/udp']
    }
}
```

```
class consul (

  $docker_version          = $consul::params::docker_version,
  $docker_tcp_bind         = $consul::params::docker_tcp_bind,
  $docker_image            = $consul::params::docker_image,
  $container_hostname      = $consul::params::container_hostname,
  $consul_advertise        = $consul::params::consul_advertise,
  $consul_bootstrap_expect = $consul::params::consul_bootstrap_expect,

  ) inherits consul::params {

  include consul::install

}
```

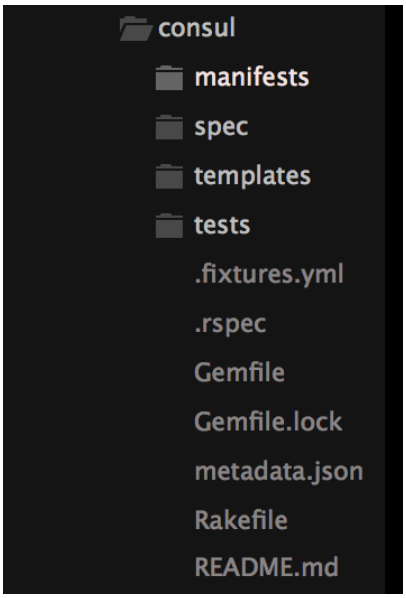
```
---
-
  box: puppetlabs/centos-7.0-64-puppet-enterprise
  cpu: 1
  ip: "172.17.8.101"
  name: node-01
  forward_ports:
     - { guest: 8500, host: 8500 }
  ram: 2048
  shell_commands:
     - { shell: 'yum install -y wget git lvm2 device-mapper-libs' }
     - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
     - { shell: 'cp /home/vagrant/node-01/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
     - { shell: cp /home/vagrant/node-01/modules/* -R /tmp/modules }
```

```
node 'node-01' {

  include consul

}
```

```
==> node-01: Running provisioner: puppet...
==> node-01: Running Puppet with default.pp...
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Notice: Compiled catalog for localhost in environment production in 1.39 seconds
==> node-01: Info: Applying configuration version '1453113319'
==> node-01: Notice: /Stage[main]/Docker::Repos/Yumrepo[docker]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/docker.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Docker::Install/Package[docker]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/Service[docker]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Docker::Service/Service[docker]: Unscheduling refresh on Service[docker]
==> node-01: Notice: /Stage[main]/Python::Install/Package[python-dev]/ensure: created
==> node-01: Notice: /Stage[main]/Epel/File[/etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7]/ensure: defined content as '{md5}58fa8ae27c89f37b08429f04fd4a88cc'
==> node-01: Notice: /Stage[main]/Epel/Epel::Rpm_gpg_key[EPEL-7]/Exec[import-EPEL-7]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Epel/Yumrepo[epel-testing]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/epel-testing.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Epel/Yumrepo[epel-testing-debuginfo]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/epel-testing-debuginfo.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Epel/Yumrepo[epel-testing-source]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/epel-testing-source.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Epel/Yumrepo[epel]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/epel.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Epel/Yumrepo[epel-debuginfo]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/epel-debuginfo.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Epel/Yumrepo[epel-source]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/epel-source.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Python::Install/Package[pip]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Docker_compose/Python::Pip[docker-compose]/Exec[pip_install_docker-compose]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Image[scottyc/consul]/Exec[docker pull scottyc/consul]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Run[consul]/File[/etc/systemd/system/docker-consul.service]/ensure: created
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[consul]/File[/etc/systemd/system/docker-consul.service]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[consul]/File[/etc/systemd/system/docker-consul.service]: Scheduling refresh of Service[docker-consul]
==> node-01: Notice: /Stage[main]/Docker::Systemd_reload/Exec[docker-systemd-reload]: Triggered 'refresh' from 2 events
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Run[consul]/Service[docker-consul]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[consul]/Service[docker-consul]: Unscheduling refresh on Service[docker-consul]
==> node-01: Notice: Finished catalog run in 73.70 seconds
```

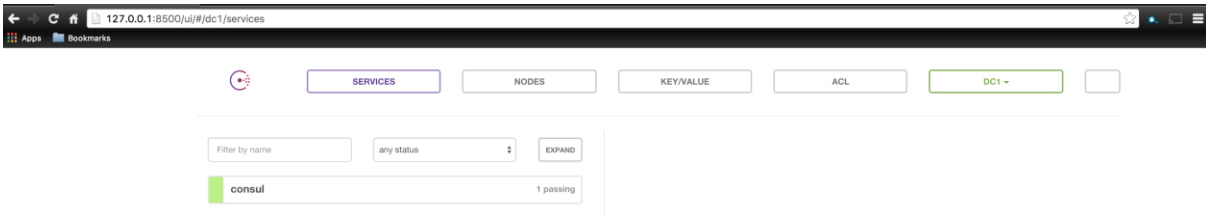


```
class consul::install {

  package { 'device-mapper-libs':
    ensure => installed,
    }

  class { 'docker':
    version     => $consul::docker_version,
    tcp_bind    => $consul::docker_tcp_bind,
    socket_bind => 'unix:///var/run/docker.sock',
    require     => Package['device-mapper-libs']
    } ->

  file { '/root/docker-compose.yml':
    ensure    => file,
    content   => template('consul/docker-compose.yml.erb'),
    } ->

  docker_compose { $consul::container_hostname :
    ensure   => present,
    source   => '/root',
    scale    => ['1']
    }
}
```



```
<%= @container_hostname %>:
    image: <%= @docker_image %>
    hostname: <%= @container_hostname %>
    ports:
      - "8300:8300"
      - "8301:8301"
      - "8301:8301/udp"
      - "8302:8302"
      - "8302:8302/udp"
      - "8400:8400"
      - "8500:8500"
      - "53:53/udp"
    command: -server --advertise <%= @consul_advertise %>  -bootstrap-expect <%= @consul_bootstrap_expect %>
```

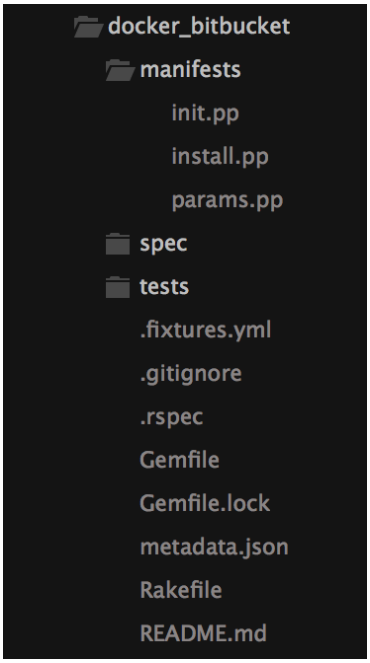
```
==> node-01: Running provisioner: puppet...
==> node-01: Running Puppet with default.pp...
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Notice: Compiled catalog for localhost in environment production in 1.29 seconds
==> node-01: Info: Applying configuration version '1453196319'
==> node-01: Notice: /Stage[main]/Docker::Repos/Yumrepo[docker]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/docker.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Docker::Install/Package[docker]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-01: Notice: /Stage[main]/Docker::Systemd_reload/Exec[docker-systemd-reload]: Triggered 'refresh' from 1 events
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/Service[docker]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Docker::Service/Service[docker]: Unscheduling refresh on Service[docker]
==> node-01: Notice: /Stage[main]/Python::Install/Package[python-dev]/ensure: created
==> node-01: Notice: /Stage[main]/Epel/File[/etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7]/ensure: defined content as '{md5}58fa8ae27c89f37b08429f04fd4a88cc'
==> node-01: Notice: /Stage[main]/Consul::Install/File[/root/docker-compose.yml]/ensure: defined content as '{md5}8bdd827a303c79ce7f65826381517d13'
==> node-01: Notice: /Stage[main]/Epel/Epel::Rpm_gpg_key[EPEL-7]/Exec[import-EPEL-7]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Epel/Yumrepo[epel-testing]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/epel-testing.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Epel/Yumrepo[epel-testing-debuginfo]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/epel-testing-debuginfo.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Epel/Yumrepo[epel-testing-source]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/epel-testing-source.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Epel/Yumrepo[epel]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/epel.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Epel/Yumrepo[epel-debuginfo]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/epel-debuginfo.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Epel/Yumrepo[epel-source]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/epel-source.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Python::Install/Package[pip]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Docker_compose/Python::Pip[docker-compose]/Exec[pip_install_docker-compose]/returns: executed successfully
==> node-01: Info: Checking if docker-compose.yml exists
==> node-01: Info: bring up containers
==> node-01: Notice: /Stage[main]/Consul::Install/Docker_compose[consul]/ensure: created
==> node-01: Notice: Finished catalog run in 75.34 seconds
```




```
[root@node-01 ~]# docker ps
CONTAINER ID   IMAGE           COMMAND                  CREATED         STATUS         PORTS                                                                                                                                    NAMES
21a72b467cb0   scottyc/consul  "/usr/sbin/consul age"   4 minutes ago   Up 4 minutes   0.0.0.0:53->53/udp, 0.0.0.0:8300-8302->8300-8302/tcp, 0.0.0.0:8400->8400/tcp, 0.0.0.0:8301-8302->8301-8302/udp, 0.0.0.0:8500->8500/tcp   root_cons
[root@node-01 ~]#
```

Chapter 4: Building Multicontainer Applications

```
EXPOSE 53/udp 8300 8301 8301/udp 8302

VOLUME ["/data"]
```




```
class docker_bitbucket {

 include docker_bitbucket::install

}
```

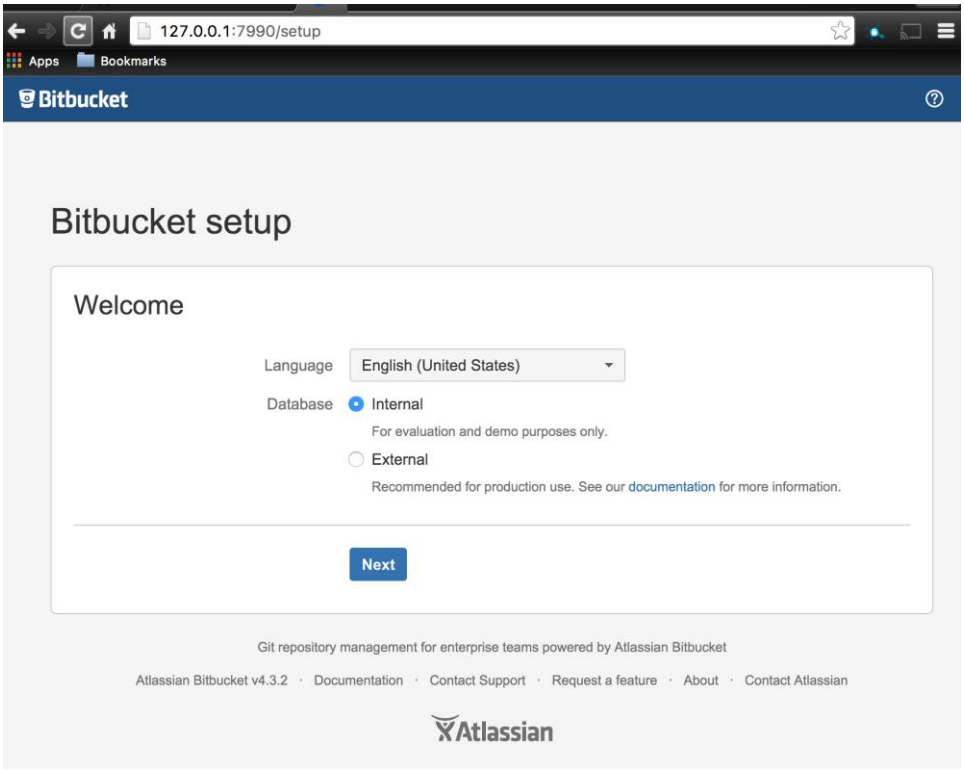
```
package { 'device-mapper-libs':
  ensure => installed,
}

class { 'docker':
  version     => '1.9.1-1.el7.centos',
  tcp_bind    => 'tcp://127.0.0.1:4243',
  socket_bind => 'unix:///var/run/docker.sock',
  require     => Package['device-mapper-libs']
} ->
```

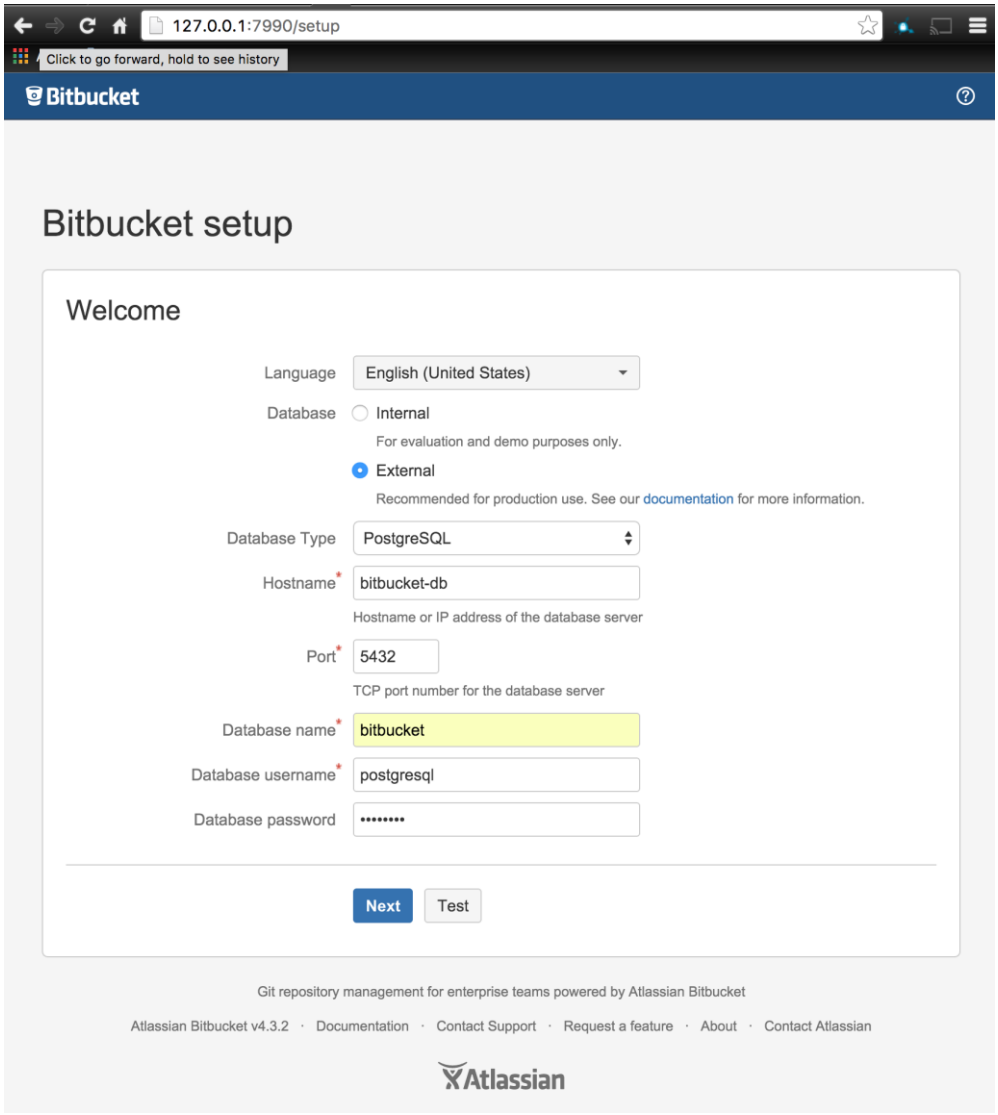
```
docker::image { 'postgres:9.2': } ->

docker::run { 'postgres':
  image    => 'postgres:9.2',
  hostname => 'bitbucket-db',
  env      => ['POSTGRES_USER=postgresql', 'POSTGRES_PASSWORD=Gr33nTe@', 'POSTGRES_DB=bitbucket', 'PGDATA=/var/lib/postgresql/data/pgdata'],
  volumes  => ['/root/db:/var/lib/postgresql/data/pgdata']
}
```

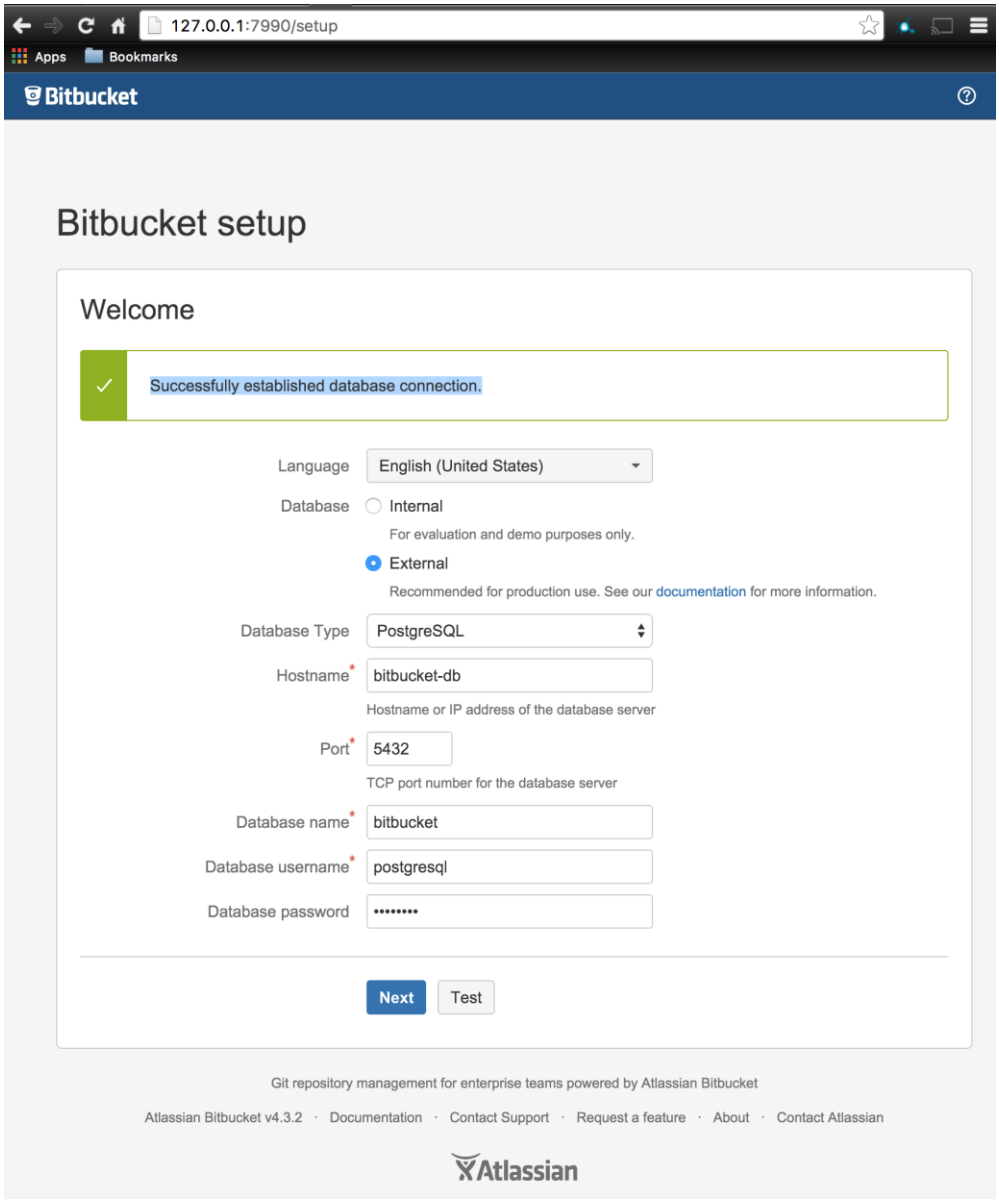
```
  docker::image { 'atlassian/bitbucket-server': } ->

  docker::run { 'bitbucket':
    image    => 'atlassian/bitbucket-server',
    ports    => ['7990:7990', '7999:7999'],
    username => 'root',
    volumes  => ['/data:/var/atlassian/application-data/bitbucket'],
    links    => ['postgres']
  }
}
```

```
---
-
  box: puppetlabs/centos-7.0-64-puppet-enterprise
  cpu: 1
  ip: "172.17.8.101"
  name: node-01
  forward_ports:
      - { guest: 7990, host: 7990 }
      - { guest: 7999, host: 7999 }
  ram: 2048
  shell_commands:
      - { shell: 'yum install -y wget git lvm2 device-mapper-libs' }
      - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
      - { shell: 'cp /home/vagrant/node-01/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
      - { shell: cp /home/vagrant/node-01/modules/* -R /tmp/modules }


```

```
    node-01: Running: inline script
==> node-01: Running provisioner: puppet...
==> node-01: Running Puppet with default.pp...
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Notice: Compiled catalog for localhost in environment production in 1.17 seconds
==> node-01: Info: Applying configuration version '1455440864'
==> node-01: Notice: /Stage[main]/Docker::Repos/Yumrepo[docker]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/docker.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Docker::Install/Package[docker]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]: Scheduling refresh of Exec[docker-systemd-reload-before-service]
==> node-01: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]: Triggered 'refresh' from 1 events
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/Service[docker]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Docker::Service/Service[docker]: Unscheduling refresh on Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Compose/Exec[Install Docker Compose 1.6.0]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose-1.6.0]/mode: mode changed '0644' to '0755'
==> node-01: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose]/ensure: created
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker::Image[postgres:9.2]/File[/usr/local/bin/update_docker_image.sh]/ensure: created
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker::Image[postgres:9.2]/Exec[/usr/local/bin/update_docker_image.sh postgres:9.2]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker::Image[atlassian/bitbucket-server]/Exec[/usr/local/bin/update_docker_image.sh atlassian/bitbucket-server]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker::Run[postgres]/File[/etc/systemd/system/docker-postgres.service]/ensure: created
==> node-01: Info: /Stage[main]/Docker_bitbucket::Install/Docker::Run[postgres]/File[/etc/systemd/system/docker-postgres.service]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-01: Info: /Stage[main]/Docker_bitbucket::Install/Docker::Run[postgres]/File[/etc/systemd/system/docker-postgres.service]: Scheduling refresh of Service[docker-postgres]
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker::Run[bitbucket]/File[/etc/systemd/system/docker-bitbucket.service]/ensure: created
==> node-01: Info: /Stage[main]/Docker_bitbucket::Install/Docker::Run[bitbucket]/File[/etc/systemd/system/docker-bitbucket.service]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-01: Info: /Stage[main]/Docker_bitbucket::Install/Docker::Run[bitbucket]/File[/etc/systemd/system/docker-bitbucket.service]: Scheduling refresh of Service[docker-bitbucket]
==> node-01: Notice: /Stage[main]/Docker::Systemd_reload/Exec[docker-systemd-reload]: Triggered 'refresh' from 2 events
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker::Run[postgres]/Service[docker-postgres]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Docker_bitbucket::Install/Docker::Run[postgres]/Service[docker-postgres]: Unscheduling refresh on Service[docker-postgres]
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker::Run[bitbucket]/Service[docker-bitbucket]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Docker_bitbucket::Install/Docker::Run[bitbucket]/Service[docker-bitbucket]: Unscheduling refresh on Service[docker-bitbucket]
==> node-01: Notice: Finished catalog run in 160.61 seconds
```


127.0.0.1:7990/setup
Apps
Bookmarks
Bitbucket
Bitbucket setup
Welcome
Language
English (United States)
Database
Internal
For evaluation and demo purposes only.
External
Recommended for production use. See our documentation for more information.
Next
Git repository management for enterprise teams powered by Atlassian Bitbucket
Atlassian Bitbucket v4.3.2 · Documentation · Contact Support · Request a feature · About · Contact Atlassian
Atlassian

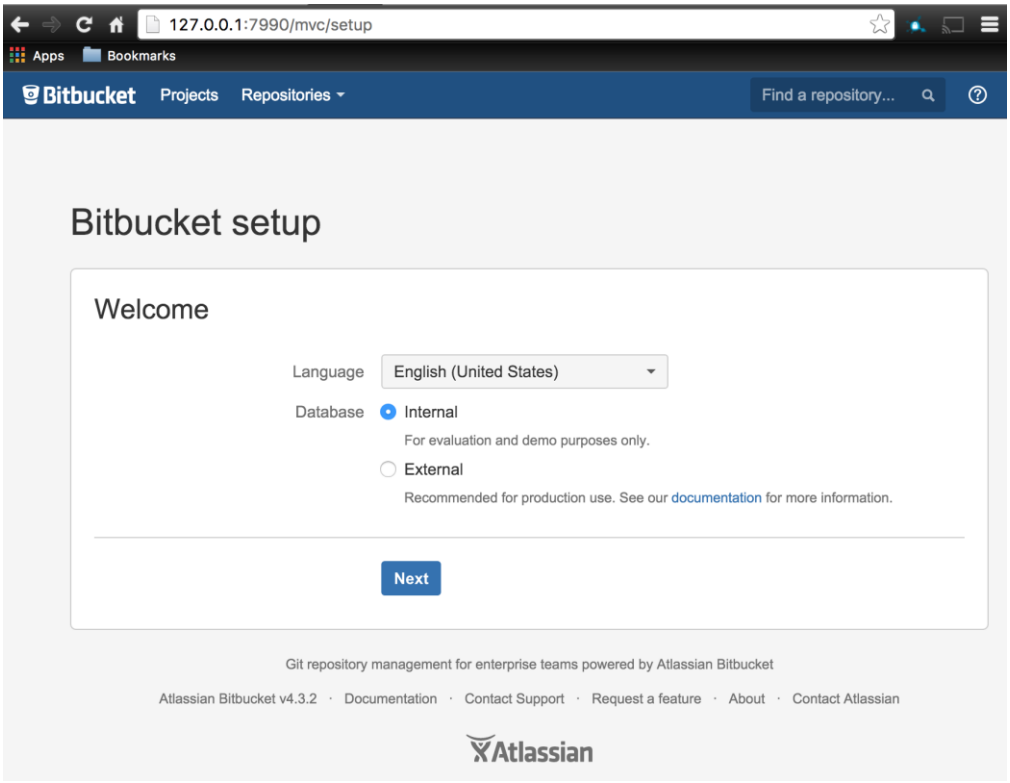

127.0.0.1:7990/setup
Click to go forward, hold to see history
Bitbucket
Bitbucket setup
Welcome
Language
English (United States)
Database
Internal
For evaluation and demo purposes only.
External
Recommended for production use. See our documentation for more information.
Database Type
PostgreSQL
Hostname*
bitbucket-db
Hostname or IP address of the database server
Port*
5432
TCP port number for the database server
Database name*
bitbucket
Database username*
postgresql
Database password
Next
Test
Git repository management for enterprise teams powered by Atlassian Bitbucket
Atlassian Bitbucket v4.3.2 · Documentation · Contact Support · Request a feature · About · Contact Atlassian
Atlassian


127.0.0.1:7990/setup
Apps
Bookmarks
Bitbucket
Bitbucket setup
Welcome
Successfully established database connection.
Language
English (United States)
Database
Internal
For evaluation and demo purposes only.
External
Recommended for production use. See our documentation for more information.
Database Type
PostgreSQL
Hostname*
bitbucket-db
Hostname or IP address of the database server
Port*
5432
TCP port number for the database server
Database name*
bitbucket
Database username*
postgresql
Database password
Next
Test
Git repository management for enterprise teams powered by Atlassian Bitbucket
Atlassian Bitbucket v4.3.2 · Documentation · Contact Support · Request a feature · About · Contact Atlassian
Atlassian

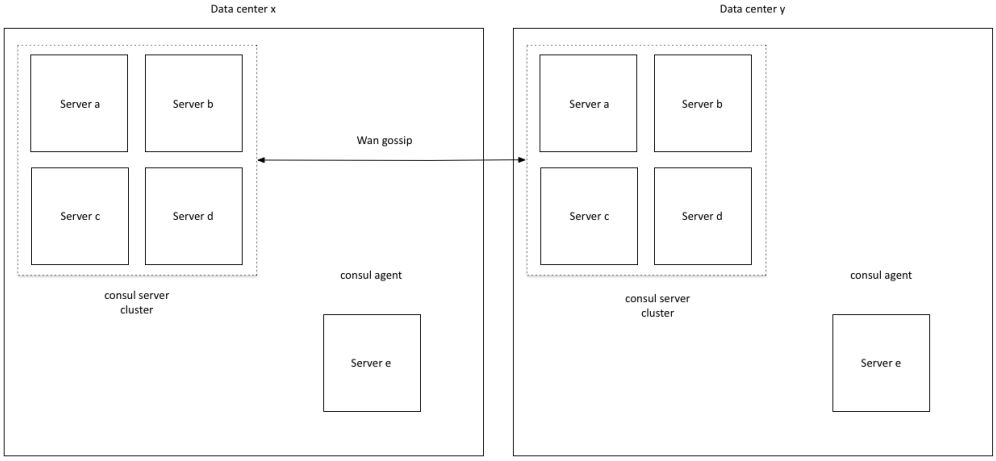
```
class docker_bitbucket::install {

  package { 'device-mapper-libs':
    ensure => installed,
    }

  class { 'docker':
    version     => '1.9.1-1.el7.centos',
    tcp_bind    => 'tcp://127.0.0.1:4243',
    socket_bind => 'unix:///var/run/docker.sock',
    require     => Package['device-mapper-libs']
  } ->

  file { '/root/docker-compose.yml':
    ensure    => file,
    content   => template('docker_bitbucket/docker-compose.yml.erb'),
    } ->

  docker_compose { 'bitbucket' :
    ensure   => present,
    source   => '/root',
    scale    => ['1']
    }
}
```

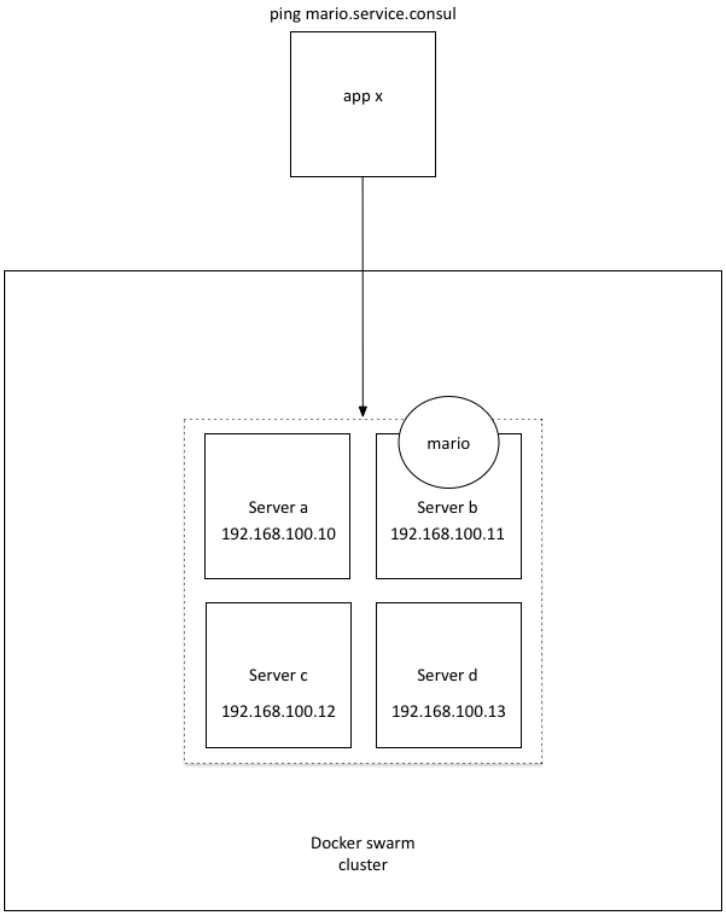
```
postgres:
    image: postgres:9.2
    environment:
        - POSTGRES_USER=postgresql
        - POSTGRES_PASSWORD=Gr33nTe@
        - POSTGRES_DB=bitbucket
        - PGDATA=/var/lib/postgresql/data/pgdata
    volumes:
        - ./db:/var/lib/postgresql/data/pgdata
bitbucket:
    image: atlassian/bitbucket-server
    user: root
    ports:
        - "7990:7990"
        - "7999:7999"
    volumes:
        - ./data:/var/atlassian/application-data/bitbucket atlassian/bitbucket-server chown -R daemon /var/atlassian/application-data/bitbucket
        - ./data:/var/atlassian/application-data/bitbucket
    links:
        - postgres
```

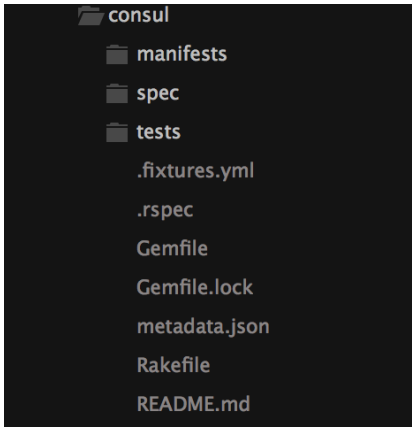
```
==> node-01: Running provisioner: puppet...
==> node-01: Running Puppet with default.pp...
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Notice: Compiled catalog for localhost in environment production in 1.03 seconds
==> node-01: Info: Applying configuration version '1455445653'
==> node-01: Notice: /Stage[main]/Docker::Repos/Yumrepo[docker]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/docker.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Docker::Install/Package[docker]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]: Scheduling refresh of Exec[docker-systemd-reload-before-service]
==> node-01: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]: Triggered 'refresh' from 1 events
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/Service[docker]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Docker::Service/Service[docker]: Unscheduling refresh on Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Compose/Exec[Install Docker Compose 1.6.0]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose-1.6.0]/mode: mode changed '0644' to '0755'
==> node-01: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose]/ensure: created
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/File[/root/docker-compose.yml]/ensure: defined content as '{md5}c6c19b0c5b81776d46401734d67c610f'
==> node-01: Info: Checking if docker-compose.yml exists
==> node-01: Info: bring up containers
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker_compose[bitbucket]/ensure: created
==> node-01: Notice: Finished catalog run in 166.21 seconds
```



Chapter 5: Configuring Service Discovery and Docker Networking




ping mario.service.consul
app x
mario
Server a
192.168.100.10
Server b
192.168.100.11
Server c
192.168.100.12
Server d
192.168.100.13
Docker swarm
cluster




```
class consul::install {

  package { 'device-mapper-libs':
    ensure => installed,
    }

  class { 'docker':
    version     => $consul::docker_version,
    tcp_bind    => $consul::docker_tcp_bind,
    socket_bind => 'unix:///var/run/docker.sock',
    require     => Package['device-mapper-libs']
    } ->

  docker::image { $consul::docker_image : } ->

  docker::run { $consul::container_hostname:
    image    => $consul::docker_image,
    hostname => $consul::container_hostname,
    command  => "-server --advertise ${consul::consul_advertise} -bootstrap-expect ${consul::consul_bootstrap_expect}",
    ports    => ['8301:8301', '8301:8301/udp', '8302:8302', '8302:8302/udp', '8400:8400', '8500:8500', '53:53/udp']
    }
}
```

```
class consul::params {

  $docker_version          = '1.9.1-1.el7.centos'
  $docker_tcp_bind         = 'tcp://127.0.0.1:4243'
  $docker_image            = 'scottyc/consul'
  $container_hostname      = 'consul'
  $consul_advertise        = $::ipaddress_enp0s8
  $consul_bootstrap_expect = '1'
}

```

```
class consul::params {

  $docker_version             = '1.9.1-1.el7.centos'
  $docker_tcp_bind            = 'tcp://127.0.0.1:4243'
  $consul_docker_image        = 'scottyc/consul'
  $consul_container_hostname  = 'consul'
  $consul_advertise           = $::ipaddress_enp0s8
  $consul_bootstrap_expect    = '1'
}

```

```
class consul::params {

  $docker_version             = '1.9.1-1.el7.centos'
  $docker_tcp_bind            = 'tcp://127.0.0.1:4243'
  $consul_docker_image        = 'scottyc/consul'
  $consul_container_hostname  = 'consul'
  $consul_advertise           = $::ipaddress_enp0s8
  $consul_bootstrap_expect    = '1'
  $reg_docker_image           = 'gliderlabs/registrator'
  $reg_container_hostname     = 'registrator'
  $reg_net                    = 'host'
  $reg_volume                 = ['/var/run/docker.sock:/tmp/docker.sock']
  $reg_command                = "consul://$::ipaddress_enp0s8:8500"
}

```

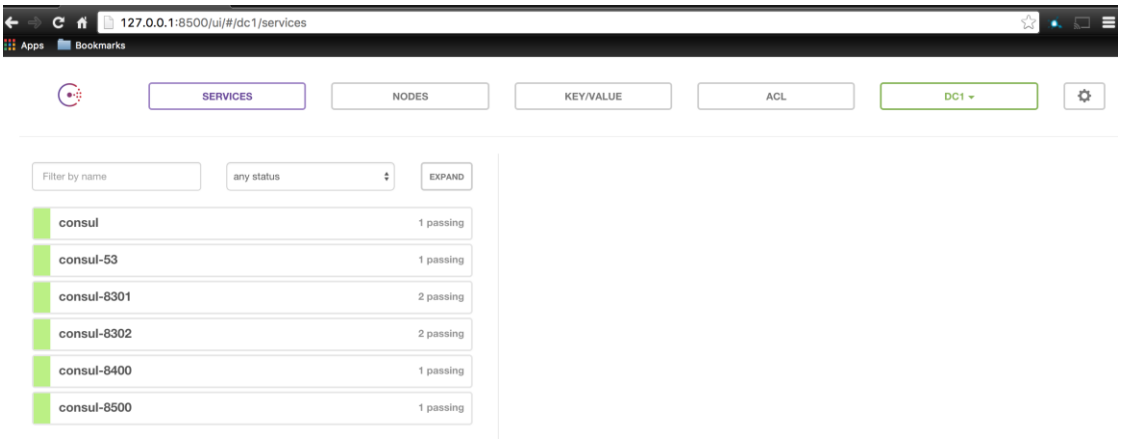
```
$docker_version             = $consul::params::docker_version,
$docker_tcp_bind            = $consul::params::docker_tcp_bind,
$consul_docker_image        = $consul::params::consul_docker_image,
$consul_container_hostname  = $consul::params::consul_container_hostname,
$consul_advertise           = $consul::params::consul_advertise,
$consul_bootstrap_expect    = $consul::params::consul_bootstrap_expect,
```

```
#
class consul (

  $docker_version            = $consul::params::docker_version,
  $docker_tcp_bind           = $consul::params::docker_tcp_bind,
  $consul_docker_image       = $consul::params::consul_docker_image,
  $consul_container_hostname = $consul::params::consul_container_hostname,
  $consul_advertise          = $consul::params::consul_advertise,
  $consul_bootstrap_expect   = $consul::params::consul_bootstrap_expect,
  $reg_docker_image          = $consul::params::reg_docker_image,
  $reg_container_hostname    = $consul::params::reg_container_hostname,
  $reg_net                   = $consul::params::reg_net,
  $reg_volume                = $consul::params::reg_volume,
  $reg_command               = $consul::params::reg_command,
```

```

  docker::image { $consul::reg_docker_image : } ->

  docker::run { $consul::reg_container_hostname:
    image    => $consul::reg_docker_image,
    net      => $consul::reg_net,
    volumes  => $consul::reg_volume,
    command  => $consul::reg_command,
    }
}


```

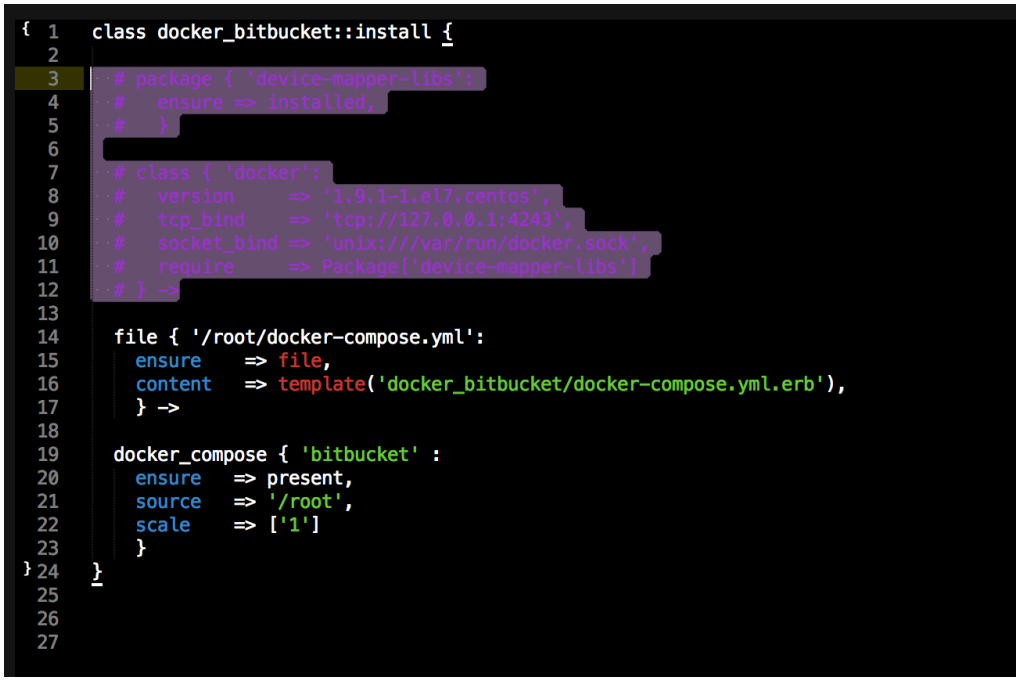
```
---
-
  box: puppetlabs/centos-7.0-64-puppet-enterprise
  cpu: 1
  ip: "172.17.8.101"
  name: node-01
  forward_ports:
      - { guest: 8500, host: 8500 }
  ram: 2048
  shell_commands:
      - { shell: 'yum install -y wget git lvm2 device-mapper-libs' }
      - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
      - { shell: 'cp /home/vagrant/node-01/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
      - { shell: cp /home/vagrant/node-01/modules/* -R /tmp/modules }


```

```
==> node-01: Running Puppet with default.pp...
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Notice: Compiled catalog for localhost in environment production in 1.22 seconds
==> node-01: Info: Applying configuration version '1456394097'
==> node-01: Notice: /Stage[main]/Docker::Repos/Yumrepo[docker]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/docker.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Docker::Install/Package[docker]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]: Scheduling refresh of Exec[docker-systemd-reload-before-service]
==> node-01: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]: Triggered 'refresh' from 1 events
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/Service[docker]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Docker::Service/Service[docker]: Unscheduling refresh on Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Compose/Exec[Install Docker Compose 1.6.0]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose-1.6.0]/mode: mode changed '0644' to '0755'
==> node-01: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose]/ensure: created
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Image[scottyc/consul]/File[/usr/local/bin/update_docker_image.sh]/ensure: created
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Image[scottyc/consul]/Exec[/usr/local/bin/update_docker_image.sh scottyc/consul]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Image[gliderlabs/registrator]/Exec[/usr/local/bin/update_docker_image.sh gliderlabs/registrator]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Run[consul]/File[/etc/systemd/system/docker-consul.service]/ensure: created
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[consul]/File[/etc/systemd/system/docker-consul.service]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[consul]/File[/etc/systemd/system/docker-consul.service]: Scheduling refresh of Service[docker-consul]
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Run[registrator]/File[/etc/systemd/system/docker-registrator.service]/ensure: created
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[registrator]/File[/etc/systemd/system/docker-registrator.service]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[registrator]/File[/etc/systemd/system/docker-registrator.service]: Scheduling refresh of Service[docker-registrator]
==> node-01: Notice: /Stage[main]/Docker::Systemd_reload/Exec[docker-systemd-reload]: Triggered 'refresh' from 2 events
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Run[consul]/Service[docker-consul]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[consul]/Service[docker-consul]: Unscheduling refresh on Service[docker-consul]
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Run[registrator]/Service[docker-registrator]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[registrator]/Service[docker-registrator]: Unscheduling refresh on Service[docker-registrator]
==> node-01: Notice: Finished catalog run in 146.49 seconds
```



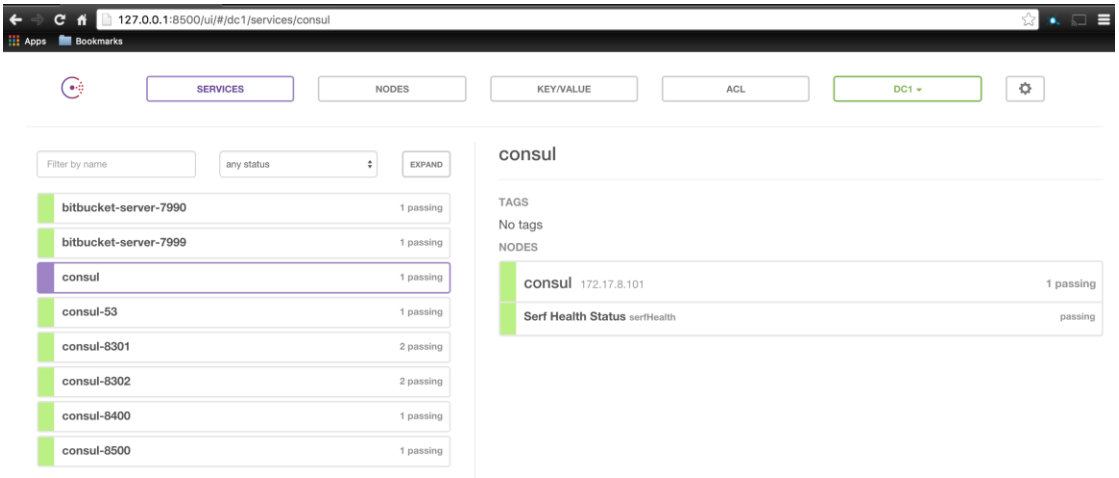
```
node 'node-01' {

  include consul
  include docker_bitbucket

}
```

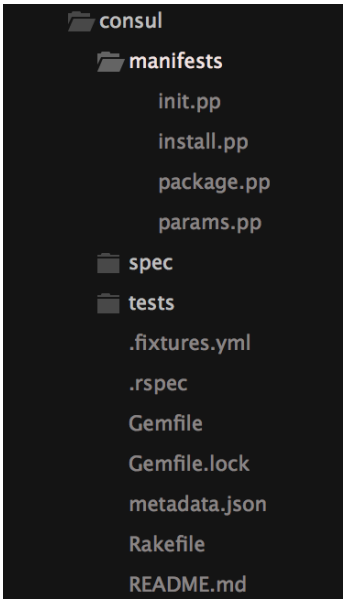
```
class docker_bitbucket::install {

  # package { 'device-mapper-libs':
  #   ensure => installed,
  # }

  # class { 'docker':
  #   version     => '1.9.1-1.el7.centos',
  #   tcp_bind    => 'tcp://127.0.0.1:4243',
  #   socket_bind => 'unix:///var/run/docker.sock',
  #   require     => Package['device-mapper-libs']
  # } ->

  docker::image { 'postgres:9.2': } ->

  docker::run { 'postgres':
    image    => 'postgres:9.2',
    hostname => 'bitbucket-db',
    env      => ['POSTGRES_USER=postgresql', 'POSTGRES_PASSWORD=Gr33nTe@', 'POSTGRES_DB=bitbucket', 'PGDATA=/var/lib/postgresql/data/pgdata'],
    volumes  => ['/root/db:/var/lib/postgresql/data/pgdata']
  }

  docker::image { 'atlassian/bitbucket-server': } ->

  docker::run { 'bitbucket':
    image    => 'atlassian/bitbucket-server',
    username => 'root',
    ports    => ['7990:7990', '7999:7999'],
    volumes  => ['/data:/var/atlassian/application-data/bitbucket'],
    links    => ['postgres']
  }
}
```



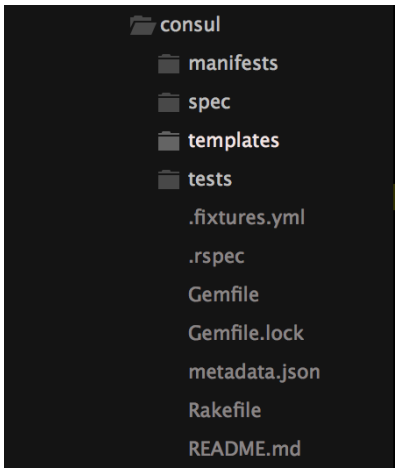
```
class docker_bitbucket::install {

  # package { 'device-mapper-libs':
  #   ensure => installed,
  #   }

  # class { 'docker':
  #   version     => '1.9.1-1.el7.centos',
  #   tcp_bind    => 'tcp://127.0.0.1:4243',
  #   socket_bind => 'unix:///var/run/docker.sock',
  #   require     => Package['device-mapper-libs']
  # } ->

  file { '/root/docker-compose.yml':
    ensure    => file,
    content   => template('docker_bitbucket/docker-compose.yml.erb'),
    } ->

  docker_compose { 'bitbucket' :
    ensure   => present,
    source   => '/root',
    scale    => ['1']
    }
}
```

```
==> node-01: Running Puppet with default.pp...
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Notice: Compiled catalog for localhost in environment production in 1.18 seconds
==> node-01: Info: Applying configuration version '1456398403'
==> node-01: Notice: /Stage[main]/Docker::Repos/Yumrepo[docker]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/docker.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Docker::Install/Package[docker]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]: Scheduling refresh of Exec[docker-systemd-reload-before-service]
==> node-01: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]: Triggered 'refresh' from 1 events
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/Service[docker]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Docker::Service/Service[docker]: Unscheduling refresh on Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Compose/Exec[Install Docker Compose 1.6.0]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose-1.6.0]/mode: mode changed '0644' to '0755'
==> node-01: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose]/ensure: created
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/File[/root/docker-compose.yml]/ensure: defined content as '{md5}c6c19b0c5b81776d46401734d67c610f'
==> node-01: Info: Checking if docker-compose.yml exists
==> node-01: Info: bring up containers
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker_compose[bitbucket]/ensure: created
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Image[scottyc/consul]/File[/usr/local/bin/update_docker_image.sh]/ensure: created
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Image[scottyc/consul]/Exec[/usr/local/bin/update_docker_image.sh scottyc/consul]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Image[gliderlabs/registrator]/Exec[/usr/local/bin/update_docker_image.sh gliderlabs/registrator]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Run[consul]/File[/etc/systemd/system/docker-consul.service]/ensure: created
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[consul]/File[/etc/systemd/system/docker-consul.service]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[consul]/File[/etc/systemd/system/docker-consul.service]: Scheduling refresh of Service[docker-consul]
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Run[registrator]/File[/etc/systemd/system/docker-registrator.service]/ensure: created
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[registrator]/File[/etc/systemd/system/docker-registrator.service]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[registrator]/File[/etc/systemd/system/docker-registrator.service]: Scheduling refresh of Service[docker-registrator]
==> node-01: Notice: /Stage[main]/Docker::Systemd_reload/Exec[docker-systemd-reload]: Triggered 'refresh' from 2 events
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Run[consul]/Service[docker-consul]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[consul]/Service[docker-consul]: Unscheduling refresh on Service[docker-consul]
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Run[registrator]/Service[docker-registrator]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[registrator]/Service[docker-registrator]: Unscheduling refresh on Service[docker-registrator]
==> node-01: Notice: Finished catalog run in 245.82 seconds
```

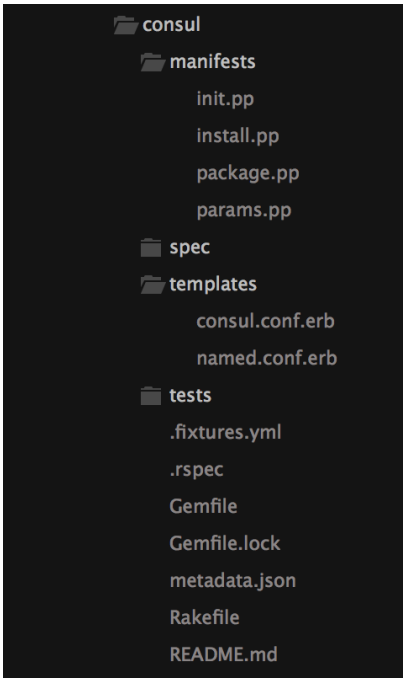



consul
manifests
init.pp
install.pp
package.pp
params.pp
spec
tests
.fixtures.yml
.rspec
Gemfile
Gemfile.lock
metadata.json
Rakefile
README.md

```
1  class consul::package {
2
3    package { 'bind':
4      ensure => present
5    } ->
6
7    file { '/etc/named.conf':
8      ensure  => present,
9      content => template("consul/named.conf.erb"),
10     mode    => '0644',
11     owner   => 'root',
12     group   => 'root',
13     require => Package['bind'],
14   } ~>
15
16   file { '/etc/named/consul.conf':
17     ensure  => present,
""18     content => template("consul/consul.conf.erb"),
19     mode    => '0644',
20     owner   => 'root',
21     group   => 'root',
22     require => Package['bind'],
23   } ~>
24
25
26   service { 'named':
27     ensure  => running,
28     enable  => true,
29     require => File['/etc/named.conf'],
30   }
31 }
```



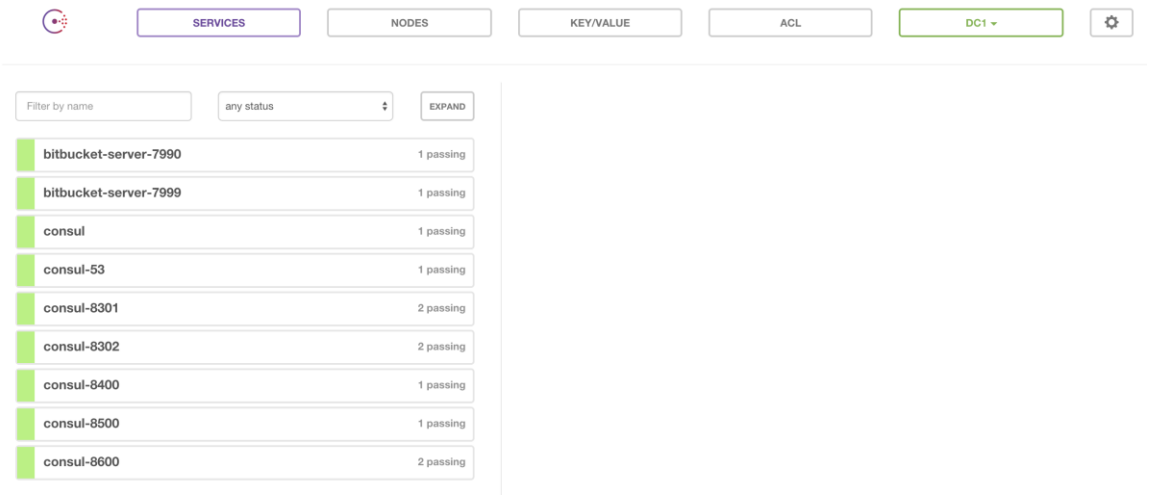
```
options {
  listen-on port 53 { 127.0.0.1; };
  listen-on-v6 port 53 { ::1; };
  directory       "/var/named";
  dump-file       "/var/named/data/cache_dump.db";
  statistics-file "/var/named/data/named_stats.txt";
  memstatistics-file "/var/named/data/named_mem_stats.txt";
  allow-query     { localhost; };
  recursion yes;

  dnssec-enable no;
  dnssec-validation no;

  /* Path to ISC DLV key */
  bindkeys-file "/etc/named.iscdlv.key";

  managed-keys-directory "/var/named/dynamic";
};

include "/etc/named/consul.conf";
```



```
zone "consul" IN {
  type forward;
  forward only;
  forwarders { 127.0.0.1 port 8600; };
};
```

```
class consul::install {

  package { 'device-mapper-libs':
    ensure => '1.02.107-5.el7_2.1',
    }

  class { 'docker':
    version     => $consul::docker_version,
    tcp_bind    => $consul::docker_tcp_bind,
    socket_bind => 'unix:///var/run/docker.sock',
    require     => Package['device-mapper-libs']
    } ->

  docker::image { $consul::consul_docker_image : } ->

  docker::run { $consul::consul_container_hostname:
    image    => $consul::consul_docker_image,
    hostname => $consul::consul_container_hostname,
    command  => "-server --advertise ${consul::consul_advertise} -bootstrap-expect ${consul::consul_bootstrap_expect}",
    ports    => ['8301:8301', '8301:8301/udp', '8302:8302', '8302:8302/udp', '8400:8400', '8500:8500', '8600:8600', '8600:8600/udp']
    }

  docker::image { $consul::reg_docker_image : } ->

  docker::run { $consul::reg_container_hostname:
    image    => $consul::reg_docker_image,
    net      => $consul::reg_net,
    volumes  => $consul::reg_volume,
    command  => $consul::reg_command,
    }
}
```

```
class consul (

  $docker_version            = $consul::params::docker_version,
  $docker_tcp_bind           = $consul::params::docker_tcp_bind,
  $consul_docker_image       = $consul::params::consul_docker_image,
  $consul_container_hostname = $consul::params::consul_container_hostname,
  $consul_advertise          = $consul::params::consul_advertise,
  $consul_bootstrap_expect   = $consul::params::consul_bootstrap_expect,
  $reg_docker_image          = $consul::params::reg_docker_image,
  $reg_container_hostname    = $consul::params::reg_container_hostname,
  $reg_net                   = $consul::params::reg_net,
  $reg_volume                = $consul::params::reg_volume,
  $reg_command               = $consul::params::reg_command,

  ) inherits consul::params {

  include consul::install
  include consul::package

}
```



```
---
-
  box: puppetlabs/centos-7.0-64-puppet-enterprise
  cpu: 1
  ip: "172.17.8.101"
  name: node-01
  forward_ports:
      - { guest: 8500, host: 8500 }
  ram: 2048
  shell_commands:
      - { shell: 'yum install -y wget git lvm2 device-mapper-libs' }
      - { shell: 'echo -e "PEERDNS=no\nDNS1=127.0.0.1\nDNS2=8.8.8.8">>/etc/sysconfig/network-scripts/ifcfg-enp0s3 && systemctl restart network'}
      - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
      - { shell: 'cp /home/vagrant/node-01/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
      - { shell: cp /home/vagrant/node-01/modules/* -R /tmp/modules }


```

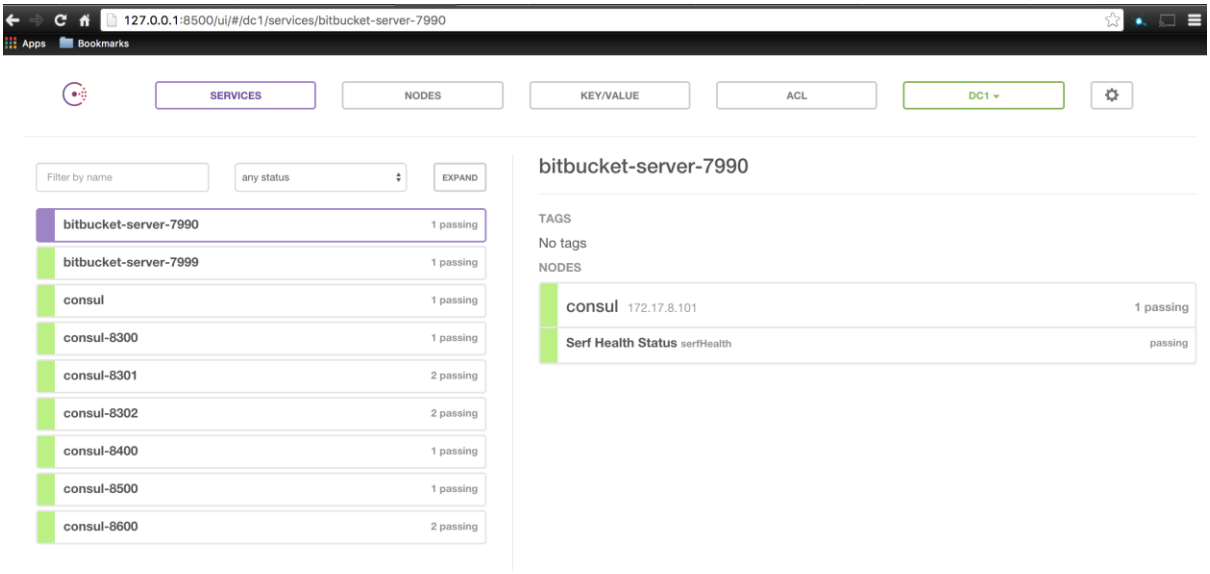
```
==> node-01: Running Puppet with default.pp...
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Notice: Compiled catalog for localhost in environment production in 1.19 seconds
==> node-01: Info: Applying configuration version '1456441982'
==> node-01: Notice: /Stage[main]/Docker::Repos/Yumrepo[docker]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/docker.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Docker::Install/Package[docker]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]: Scheduling refresh of Exec[docker-systemd-reload-before-service]
==> node-01: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]: Triggered 'refresh' from 1 events
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/Service[docker]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Docker::Service/Service[docker]: Unscheduling refresh on Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Compose/Exec[Install Docker Compose 1.6.0]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose-1.6.0]/mode: mode changed '0644' to '0755'
==> node-01: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose]/ensure: created
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/File[/root/docker-compose.yml]/ensure: defined content as '{md5}c6c19b0c5b81776d46401734d67c610f'
==> node-01: Info: Checking if docker-compose.yml exists
==> node-01: Info: bring up containers
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker_compose[bitbucket]/ensure: created
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Image[scottyc/consul]/File[/usr/local/bin/update_docker_image.sh]/ensure: created
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Image[scottyc/consul]/Exec[/usr/local/bin/update_docker_image.sh scottyc/consul]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Image[gliderlabs/registrator]/Exec[/usr/local/bin/update_docker_image.sh gliderlabs/registrator]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Run[consul]/File[/etc/systemd/system/docker-consul.service]/ensure: created
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[consul]/File[/etc/systemd/system/docker-consul.service]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[consul]/File[/etc/systemd/system/docker-consul.service]: Scheduling refresh of Service[docker-consul]
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Run[registrator]/File[/etc/systemd/system/docker-registrator.service]/ensure: created
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[registrator]/File[/etc/systemd/system/docker-registrator.service]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[registrator]/File[/etc/systemd/system/docker-registrator.service]: Scheduling refresh of Service[docker-registrator]
==> node-01: Notice: /Stage[main]/Docker::Systemd_reload/Exec[docker-systemd-reload]: Triggered 'refresh' from 2 events
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Run[consul]/Service[docker-consul]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[consul]/Service[docker-consul]: Unscheduling refresh on Service[docker-consul]
==> node-01: Notice: /Stage[main]/Consul::Install/Docker::Run[registrator]/Service[docker-registrator]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Consul::Install/Docker::Run[registrator]/Service[docker-registrator]: Unscheduling refresh on Service[docker-registrator]
==> node-01: Notice: Finished catalog run in 224.22 seconds
```

```
[vagrant@node-01 ~]$ ping bitbucket-server-7990.service.consul
PING bitbucket-server-7990.service.consul (127.0.0.1) 56(84) bytes of data.
64 bytes from node-01 (127.0.0.1): icmp_seq=1 ttl=64 time=0.000 ms
64 bytes from node-01 (127.0.0.1): icmp_seq=2 ttl=64 time=0.064 ms
64 bytes from node-01 (127.0.0.1): icmp_seq=3 ttl=64 time=0.048 ms
64 bytes from node-01 (127.0.0.1): icmp_seq=4 ttl=64 time=0.068 ms
64 bytes from node-01 (127.0.0.1): icmp_seq=5 ttl=64 time=0.114 ms
64 bytes from node-01 (127.0.0.1): icmp_seq=6 ttl=64 time=0.068 ms
^C
--- bitbucket-server-7990.service.consul ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5008ms
rtt min/avg/max/mdev = 0.000/0.060/0.114/0.034 ms
[vagrant@node-01 ~]$
```

```
1  class consul::params {
2
3    $docker_version             = '1.9.1-1.el7.centos'
4    $docker_tcp_bind            = 'tcp://127.0.0.1:4243'
5    $consul_docker_image        = 'scottyc/consul'
6    $consul_container_hostname  = 'consul'
7    $consul_advertise           = $::ipaddress_enp0s8
8    $consul_bootstrap_expect    = '1'
9    $reg_docker_image           = 'gliderlabs/registrator'
10   $reg_container_hostname     = 'registrator'
11   $reg_net                    = 'host'
12   $reg_volume                 = '/var/run/docker.sock:/tmp/docker.sock'
13   $reg_command                = "consul://$::ipaddress_enp0s8:8500"
14 }
15
```

```
<%= @consul_container_hostname %>:
    image: <%= @consul_docker_image %>
    hostname: <%= @consul_container_hostname %>
    ports:
      - "8300:8300"
      - "8301:8301"
      - "8301:8301/udp"
      - "8302:8302"
      - "8302:8302/udp"
      - "8400:8400"
      - "8500:8500"
      - "8600:8600"
      - "8600:8600/udp"
    command: -server --advertise <%= @consul_advertise %>  -bootstrap-expect <%= @consul_bootstrap_expect %>

registrator:
    image: <%= @reg_docker_image %>
    net: "<%= @reg_net %>"
    volumes:
      - <%= @reg_volume %>
    command: "<%= @reg_command %>"
```

```
==> node-01: Running Puppet with default.pp...
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Notice: Compiled catalog for localhost in environment production in 1.17 seconds
==> node-01: Info: Applying configuration version '1456460640'
==> node-01: Notice: /Stage[main]/Docker::Repos/Yumrepo[docker]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/docker.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Docker::Install/Package[docker]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]: Scheduling refresh of Exec[docker-systemd-reload-before-service]
==> node-01: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]: Triggered 'refresh' from 1 events
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/Service[docker]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Docker::Service/Service[docker]: Unscheduling refresh on Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Compose/Exec[Install Docker Compose 1.6.0]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose-1.6.0]/mode: mode changed '0644' to '0755'
==> node-01: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose]/ensure: created
==> node-01: Notice: /Stage[main]/Consul::Install/File[/root/docker-compose.yml]/ensure: defined content as '{md5}16627c8400a1dcbd19c6967a03282a9f'
==> node-01: Info: Checking if docker-compose.yml exists
==> node-01: Info: bring up containers
==> node-01: Notice: /Stage[main]/Consul::Install/Docker_compose[undef]/ensure: created
==> node-01: Notice: /Stage[main]/Consul::Package/Package[bind]/ensure: created
==> node-01: Info: Computing checksum on file /etc/named.conf
==> node-01: Info: /Stage[main]/Consul::Package/File[/etc/named.conf]: Filebucketed /etc/named.conf to puppet with sum 0d049950431ec70b91d3b56cf33bb383
==> node-01: Notice: /Stage[main]/Consul::Package/File[/etc/named.conf]/content: content changed '{md5}0d049950431ec70b91d3b56cf33bb383' to '{md5}15fa26b437cf7587b8dd7d53bce7d788'
==> node-01: Notice: /Stage[main]/Consul::Package/File[/etc/named.conf]/group: group changed 'named' to 'root'
==> node-01: Notice: /Stage[main]/Consul::Package/File[/etc/named.conf]/mode: mode changed '0640' to '0644'
==> node-01: Notice: /Stage[main]/Consul::Package/File[/etc/named/consul.conf]/ensure: created
==> node-01: Info: /Stage[main]/Consul::Package/File[/etc/named/consul.conf]: Scheduling refresh of Service[named]
==> node-01: Notice: /Stage[main]/Consul::Package/Service[named]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Consul::Package/Service[named]: Unscheduling refresh on Service[named]
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker::Image[postgres:9.2]/File[/usr/local/bin/update_docker_image.sh]/ensure: created
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker::Image[postgres:9.2]/Exec[/usr/local/bin/update_docker_image.sh postgres:9.2]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker::Image[atlassian/bitbucket-server]/Exec[/usr/local/bin/update_docker_image.sh atlassian/bitbucket-server]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker::Run[postgres]/File[/etc/systemd/system/docker-postgres.service]/ensure: created
==> node-01: Info: /Stage[main]/Docker_bitbucket::Install/Docker::Run[postgres]/File[/etc/systemd/system/docker-postgres.service]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-01: Info: /Stage[main]/Docker_bitbucket::Install/Docker::Run[postgres]/File[/etc/systemd/system/docker-postgres.service]: Scheduling refresh of Service[docker-postgres]
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker::Run[bitbucket]/File[/etc/systemd/system/docker-bitbucket.service]/ensure: created
==> node-01: Info: /Stage[main]/Docker_bitbucket::Install/Docker::Run[bitbucket]/File[/etc/systemd/system/docker-bitbucket.service]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-01: Info: /Stage[main]/Docker_bitbucket::Install/Docker::Run[bitbucket]/File[/etc/systemd/system/docker-bitbucket.service]: Scheduling refresh of Service[docker-bitbucket]
==> node-01: Notice: /Stage[main]/Docker::Systemd_reload/Exec[docker-systemd-reload]: Triggered 'refresh' from 2 events
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker::Run[postgres]/Service[docker-postgres]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Docker_bitbucket::Install/Docker::Run[postgres]/Service[docker-postgres]: Unscheduling refresh on Service[docker-postgres]
==> node-01: Notice: /Stage[main]/Docker_bitbucket::Install/Docker::Run[bitbucket]/Service[docker-bitbucket]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Docker_bitbucket::Install/Docker::Run[bitbucket]/Service[docker-bitbucket]: Unscheduling refresh on Service[docker-bitbucket]
==> node-01: Notice: Finished catalog run in 196.64 seconds
```




```
[vagrant@node-01 ~]$ ping consul-8500.service.consul
PING consul-8500.service.consul (127.0.0.1) 56(84) bytes of data.
64 bytes from node-01 (127.0.0.1): icmp_seq=1 ttl=64 time=0.040 ms
64 bytes from node-01 (127.0.0.1): icmp_seq=2 ttl=64 time=0.069 ms
64 bytes from node-01 (127.0.0.1): icmp_seq=3 ttl=64 time=0.123 ms
64 bytes from node-01 (127.0.0.1): icmp_seq=4 ttl=64 time=0.051 ms
^C
--- consul-8500.service.consul ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3000ms
rtt min/avg/max/mdev = 0.040/0.070/0.123/0.033 ms
[vagrant@node-01 ~]$
```

```
6
7   class { 'docker':
8     version     => $consul::docker_version,
9     tcp_bind    => $consul::docker_tcp_bind,
10    socket_bind => 'unix:///var/run/docker.sock',
11    extra_parameters => '--cluster-store=consul://127.0.0.1:8500 --cluster-advertise=enp0s8:2376',
12    require     => Package['device-mapper-libs']
13    } ->
14
```

```
class consul::network {

   docker_network { 'docker-internal':
    ensure   => present,
    create   => true,
    driver   => 'overlay',
    }
}
```

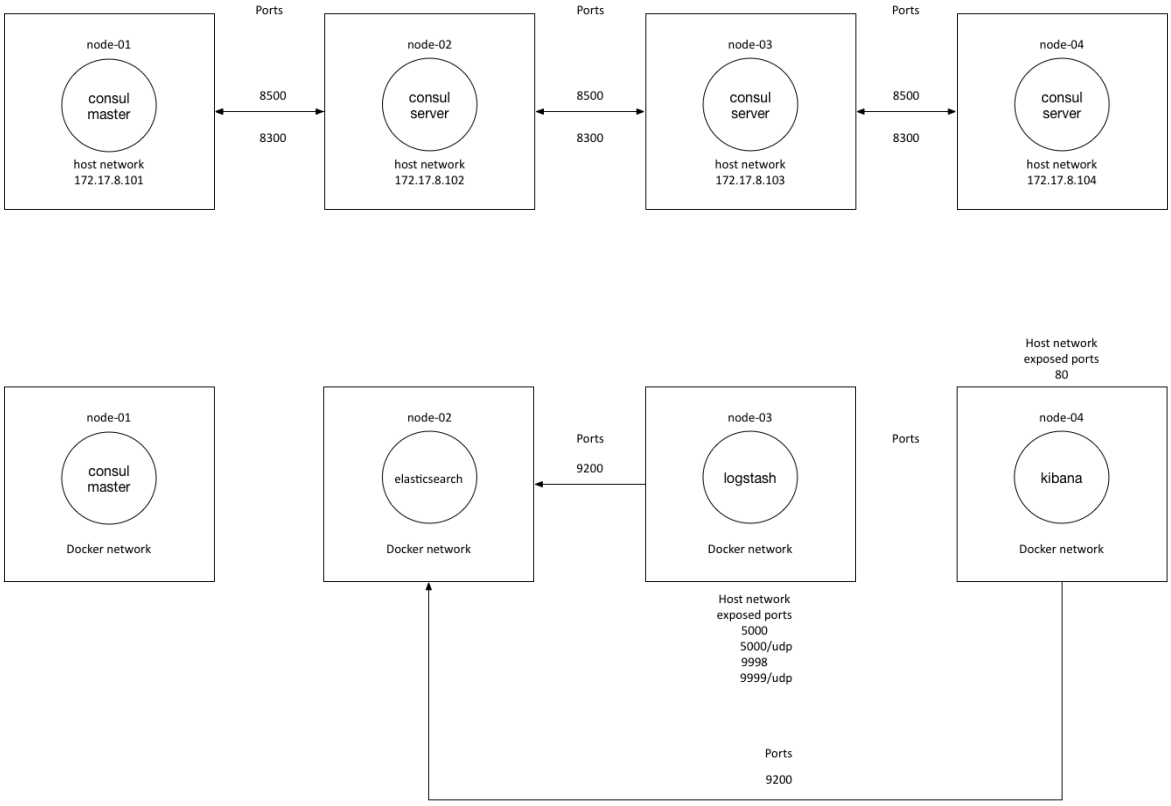
```
class consul (

  $docker_version            = $consul::params::docker_version,
  $docker_tcp_bind           = $consul::params::docker_tcp_bind,
  $consul_docker_image       = $consul::params::consul_docker_image,
  $consul_container_hostname = $consul::params::consul_container_hostname,
  $consul_advertise          = $consul::params::consul_advertise,
  $consul_bootstrap_expect   = $consul::params::consul_bootstrap_expect,
  $reg_docker_image          = $consul::params::reg_docker_image,
  $reg_container_hostname    = $consul::params::reg_container_hostname,
  $reg_net                   = $consul::params::reg_net,
  $reg_volume                = $consul::params::reg_volume,
  $reg_command               = $consul::params::reg_command,

  ) inherits consul::params {

  contain consul::install
  contain consul::package
  contain consul::network

  Class['consul::install'] -> Class['consul::package'] -> Class['consul::network']

}
```

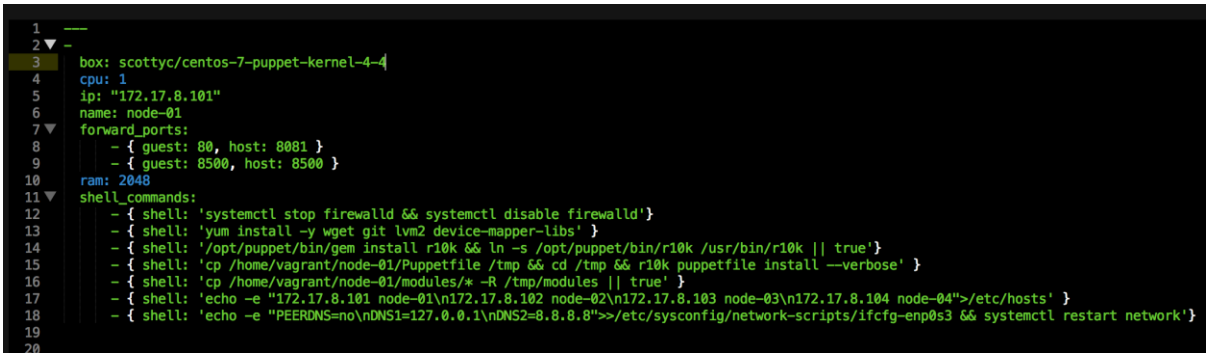
```
45  docker_network { 'my-net':
46    ensure  => present,
47    create  => true,
48    driver  => 'overlay',
49    subnet  => '192.168.1.0/24',
50    gateway => '192.168.1.1',
51    iprange => ' 192.168.1.4/32'
52  }
```

```
==> node-01: Running provisioner: puppet...
==> node-01: Running Puppet with default.pp...
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Info: Loading facts
==> node-01: Notice: Compiled catalog for localhost in environment production in 1.04 seconds
==> node-01: Info: Applying configuration version '1456696859'
==> node-01: Notice: /Stage[main]/Docker::Repos/Yumrepo[docker]/ensure: created
==> node-01: Info: changing mode of /etc/yum.repos.d/docker.repo from 600 to 644
==> node-01: Notice: /Stage[main]/Docker::Install/Package[docker]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d]/ensure: created
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]: Scheduling refresh of Exec[docker-systemd-reload-before-service]
==> node-01: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]: Triggered 'refresh' from 1 events
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]/ensure: created
==> node-01: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]: Scheduling refresh of Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Service/Service[docker]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Docker::Service/Service[docker]: Unscheduling refresh on Service[docker]
==> node-01: Notice: /Stage[main]/Docker::Compose/Exec[Install Docker Compose 1.6.0]/returns: executed successfully
==> node-01: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose-1.6.0]/mode: mode changed '0644' to '0755'
==> node-01: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose]/ensure: created
==> node-01: Notice: /Stage[main]/Consul::Install/File[/root/docker-compose.yml]/ensure: defined content as '{md5}166277c8400a1dcbd19c6967a03282a9f'
==> node-01: Info: Checking if docker-compose.yml exists
==> node-01: Info: bring up containers
==> node-01: Notice: /Stage[main]/Consul::Install/Docker_compose[undef]/ensure: created
==> node-01: Notice: /Stage[main]/Consul::Package/Package[bind]/ensure: created
==> node-01: Info: Computing checksum on file /etc/named.conf
==> node-01: Info: /Stage[main]/Consul::Package/File[/etc/named.conf]: Filebucketed /etc/named.conf to puppet with sum 0d049950431ec70b91d3b56cf33bb383
==> node-01: Notice: /Stage[main]/Consul::Package/File[/etc/named.conf]/content: content changed '{md5}0d049950431ec70b91d3b56cf33bb383' to '{md5}15fa26b437cf7587b8dd7d53bce7d788'
==> node-01: Notice: /Stage[main]/Consul::Package/File[/etc/named.conf]/group: group changed 'named' to 'root'
==> node-01: Notice: /Stage[main]/Consul::Package/File[/etc/named.conf]/mode: mode changed '0640' to '0644'
==> node-01: Notice: /Stage[main]/Consul::Package/File[/etc/named/consul.conf]/ensure: created
==> node-01: Info: /Stage[main]/Consul::Package/File[/etc/named/consul.conf]: Scheduling refresh of Service[named]
==> node-01: Notice: /Stage[main]/Consul::Package/Service[named]/ensure: ensure changed 'stopped' to 'running'
==> node-01: Info: /Stage[main]/Consul::Package/Service[named]: Unscheduling refresh on Service[named]
==> node-01: Info: checking if docker network exists
==> node-01: Info: configuring network
==> node-01: Notice: /Stage[main]/Consul::Network/Docker_network[docker-internal]/ensure: created
==> node-01: Notice: Finished catalog run in 146.78 seconds
```

```
[root@node-01 ~]# docker network ls
NETWORK ID          NAME                DRIVER
de325bfc5fd4        docker-internal     overlay
3c92cd3acc0e        bridge              bridge
8263a08be627        none                null
ebc96a1f7545        host                host
```

Chapter 6: Multinode Applications





```
---
-
  box: scottyc/centos-7-puppet-kernel-4-4
  cpu: 1
  ip: "172.17.8.101"
  name: node-01
  forward_ports:
      - { guest: 80, host: 8081 }
      - { guest: 8500, host: 8500 }
  ram: 2048
  shell_commands:
      - { shell: 'systemctl stop firewalld && systemctl disable firewalld'}
      - { shell: 'yum install -y wget git lvm2 device-mapper-libs' }
      - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
      - { shell: 'cp /home/vagrant/node-01/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
      - { shell: 'cp /home/vagrant/node-01/modules/* -R /tmp/modules || true' }
      - { shell: 'echo -e "172.17.8.101 node-01\n172.17.8.102 node-02\n172.17.8.103 node-03\n172.17.8.104 node-04">/etc/hosts' }
      - { shell: 'echo -e "PEERDNS=no\nDNS1=127.0.0.1\nDNS2=8.8.8.8">>/etc/sysconfig/network-scripts/ifcfg-enp0s3 && systemctl restart network'}

```

```yaml
---
-
  box: scottyc/centos-7-puppet-kernel-4-4
  cpu: 1
  ip: "172.17.8.101"
  name: node-01
  forward_ports:
      - { guest: 80, host: 8081 }
      - { guest: 8500, host: 8500 }
  ram: 2048
  shell_commands:
      - { shell: 'systemctl stop firewalld && systemctl disable firewalld'}
      - { shell: 'yum install -y wget git lvm2 device-mapper-libs' }
      - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
      - { shell: 'cp /home/vagrant/node-01/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
      - { shell: 'cp /home/vagrant/node-01/modules/* -R /tmp/modules || true' }
      - { shell: 'echo -e "172.17.8.101 node-01\n172.17.8.102 node-02\n172.17.8.103 node-03\n172.17.8.104 node-04">/etc/hosts' }
      - { shell: 'echo -e "PEERDNS=no\nDNS1=127.0.0.1\nDNS2=8.8.8.8">>/etc/sysconfig/network-scripts/ifcfg-enp0s3 && systemctl restart network'}


-
  box: scottyc/centos-7-puppet-kernel-4-4
  cpu: 1
  ip: "172.17.8.102"
  name: node-02
  forward_ports:
      - { guest: 80, host: 8082 }
  ram: 2048
  shell_commands:
      - { shell: 'systemctl stop firewalld && systemctl disable firewalld'}
      - { shell: 'yum install -y wget git lvm2 device-mapper-libs' }
      - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
      - { shell: 'cp /home/vagrant/node-02/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
      - { shell: 'cp /home/vagrant/node-02/modules/* -R /tmp/modules' }
      - { shell: 'echo -e "172.17.8.101 node-01\n172.17.8.102 node-02\n172.17.8.103 node-03\n172.17.8.104 node-04">/etc/hosts' }
      - { shell: 'echo -e "PEERDNS=no\nDNS1=127.0.0.1\nDNS2=8.8.8.8">>/etc/sysconfig/network-scripts/ifcfg-enp0s3 && systemctl restart network'}

-
  box: scottyc/centos-7-puppet-kernel-4-4
  cpu: 1
  ip: "172.17.8.103"
  name: node-03
  forward_ports:
      - { guest: 80, host: 8083 }
  ram: 2048
  shell_commands:
      - { shell: 'systemctl stop firewalld && systemctl disable firewalld'}
      - { shell: 'yum install -y wget git lvm2 device-mapper-libs' }
      - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
      - { shell: 'cp /home/vagrant/node-03/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
      - { shell: 'cp /home/vagrant/node-03/modules/* -R /tmp/modules' }
      - { shell: 'echo -e "172.17.8.101 node-01\n172.17.8.102 node-02\n172.17.8.103 node-03\n172.17.8.104 node-04">/etc/hosts' }
      - { shell: 'echo -e "PEERDNS=no\nDNS1=127.0.0.1\nDNS2=8.8.8.8">>/etc/sysconfig/network-scripts/ifcfg-enp0s3 && systemctl restart network'}


-
  box: scottyc/centos-7-puppet-kernel-4-4
  cpu: 1
  ip: "172.17.8.104"
  name: node-04
  forward_ports:
      - { guest: 80, host: 8080 }
  ram: 2048
  shell_commands:
      - { shell: 'systemctl stop firewalld && systemctl disable firewalld'}
      - { shell: 'yum install -y wget git lvm2 device-mapper-libs' }
      - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
      - { shell: 'cp /home/vagrant/node-04/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
      - { shell: 'cp /home/vagrant/node-04/modules/* -R /tmp/modules'}
      - { shell: 'echo -e "172.17.8.101 node-01\n172.17.8.102 node-02\n172.17.8.103 node-03\n172.17.8.104 node-04">/etc/hosts' }
      - { shell: 'echo -e "PEERDNS=no\nDNS1=127.0.0.1\nDNS2=8.8.8.8">>/etc/sysconfig/network-scripts/ifcfg-enp0s3 && systemctl restart network'}

```

```
1   class consul::params {
2
3     $docker_version             = '1.9.1-1.el7.centos'
4     $docker_tcp_bind            = 'tcp://127.0.0.1:4243'
5     $consul_docker_image        = 'scottyc/consul'
6     $consul_container_hostname  = "$hostname"
7     $consul_advertise           = $::ipaddress_enp0s8
8     $consul_bootstrap_expect    = '1'
9     $consul_master_ip           = '172.17.8.101'
10    $reg_docker_image           = 'gliderlabs/registrator'
11    $reg_container_hostname     = 'registrator'
12    $reg_net                    = 'host'
13    $reg_volume                 = '/var/run/docker.sock:/tmp/docker.sock'
14    $reg_command                = "consul://$::ipaddress_enp0s8:8500"
15
16    if ($::hostname == 'node-01') { $consul_is_master = true }
17    else { $consul_is_master = false }
18  }
19
```

```
class consul (

  $docker_version            = $consul::params::docker_version,
  $docker_tcp_bind           = $consul::params::docker_tcp_bind,
  $consul_docker_image       = $consul::params::consul_docker_image,
  $consul_container_hostname = $consul::params::consul_container_hostname,
  $consul_advertise          = $consul::params::consul_advertise,
  $consul_bootstrap_expect   = $consul::params::consul_bootstrap_expect,
  $consul_master_ip          = $consul::params::consul_master_ip,
  $consul_is_master          = $consul::params::consul_is_master,
  $reg_docker_image          = $consul::params::reg_docker_image,
  $reg_container_hostname    = $consul::params::reg_container_hostname,
  $reg_net                   = $consul::params::reg_net,
  $reg_volume                = $consul::params::reg_volume,
  $reg_command               = $consul::params::reg_command,

  ) inherits consul::params {

  validate_bool($consul_is_master)

  contain consul::install
  contain consul::package
  contain consul::network

  Class['consul::install'] -> Class['consul::package'] -> Class['consul::network']

}
```

```
class consul::install {

  package { 'device-mapper-libs':
    ensure => installed,
    }

  class { 'docker':
    version     => $consul::docker_version,
    tcp_bind    => $consul::docker_tcp_bind,
    socket_bind => 'unix:///var/run/docker.sock',
    extra_parameters => "--cluster-store=consul://$consul::consul_master_ip:8500 --cluster-advertise=enp0s8:2376",
    require     => Package['device-mapper-libs']
    } ->


  file { '/root/docker-compose.yml':
    ensure    => file,
    content   => template('consul/docker-compose.yml.erb'),
    } ->

  docker_compose { $consul::container_hostname :
    ensure   => present,
    source   => '/root',
    scale    => ['1', '1']
    }
}
```

```
class consul::network {

   docker_network { 'docker-internal':
    ensure   => present,
    create   => true,
    driver   => 'overlay',
    }
}
```

```
class consul::package {

  package { 'bind':
    ensure => present
  } ->

  file { '/etc/named.conf':
    ensure  => present,
    content => template("consul/named.conf.erb"),
    mode    => '0644',
    owner   => 'root',
    group   => 'root',
    require => Package['bind'],
  } ~>

  file { '/etc/named/consul.conf':
    ensure  => present,
    content => template("consul/consul.conf.erb"),
    mode    => '0644',
    owner   => 'root',
    group   => 'root',
    require => Package['bind'],
  } ~>


  service { 'named':
    ensure  => running,
    enable  => true,
    require => File['/etc/named.conf'],
  }
}
```

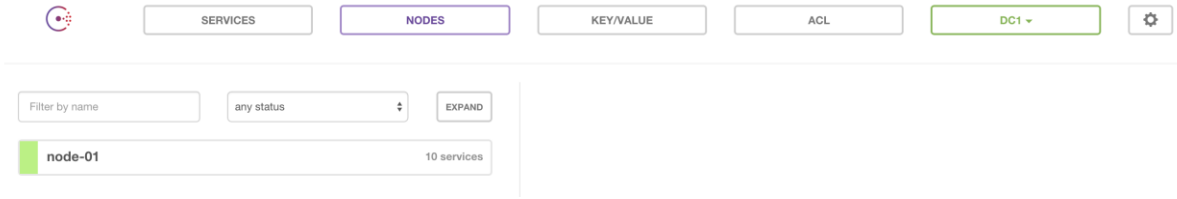
```
<% if @consul_is_master == true then -%>
<%= @consul_container_hostname %>:
    image: <%= @consul_docker_image %>
    hostname: <%= @consul_container_hostname %>
    restart: always
    ports:
      - "8300:8300"
      - "8301:8301"
      - "8301:8301/udp"
      - "8302:8302"
      - "8302:8302/udp"
      - "8400:8400"
      - "8500:8500"
      - "8600:8600"
      - "8600:8600/udp"
    command: -server --client 0.0.0.0 --advertise <%= @consul_advertise %>  -bootstrap-expect <%= @consul_bootstrap_expect %>
<% elsif @consul_is_master == false then -%>
<%= @consul_container_hostname %>:
    image: <%= @consul_docker_image %>
    hostname: <%= @consul_container_hostname %>
    ports:
      - "8300:8300"
      - "8301:8301"
      - "8301:8301/udp"
      - "8302:8302"
      - "8302:8302/udp"
      - "8400:8400"
      - "8500:8500"
      - "8600:8600"
      - "8600:8600/udp"
    command: -server -bind 0.0.0.0 --client 0.0.0.0  --advertise <%= @consul_advertise %> -join <%= @consul_master_ip %>
<% end -%>

registrator:
    image: <%= @reg_docker_image %>
    restart: always
    net: "<%= @reg_net %>"
    volumes:
      - <%= @reg_volume %>
    command: "<%= @reg_command %>"
```

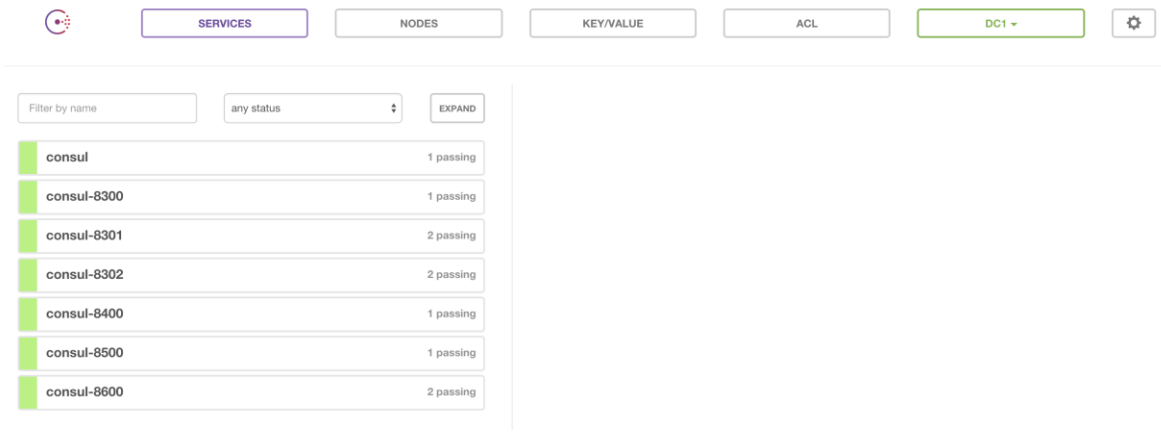
```
<% if @consul_is_master == true then -%>
<%= @consul_container_hostname %>:
    image: <%= @consul_docker_image %>
    hostname: <%= @consul_container_hostname %>
    restart: always
    ports:
      - "8300:8300"
      - "8301:8301"
      - "8301:8301/udp"
      - "8302:8302"
      - "8302:8302/udp"
      - "8400:8400"
      - "8500:8500"
      - "8600:8600"
      - "8600:8600/udp"
    command: -server --client 0.0.0.0 --advertise <%= @consul_advertise %>  -bootstrap-expect <%= @consul_bootstrap_expect %>
```

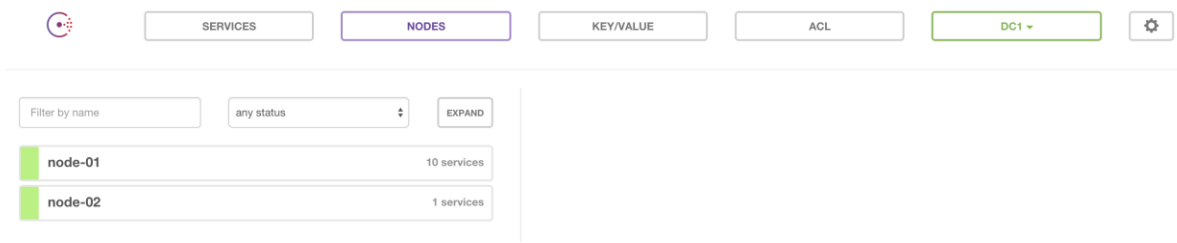
```
<% elsif @consul_is_master == false then -%>
<%= @consul_container_hostname %>:
    image: <%= @consul_docker_image %>
    hostname: <%= @consul_container_hostname %>
    ports:
      - "8300:8300"
      - "8301:8301"
      - "8301:8301/udp"
      - "8302:8302"
      - "8302:8302/udp"
      - "8400:8400"
      - "8500:8500"
      - "8600:8600"
      - "8600:8600/udp"
    command: -server -bind 0.0.0.0 --client 0.0.0.0  --advertise <%= @consul_advertise %> -join <%= @consul_master_ip %>
<% end -%>
```

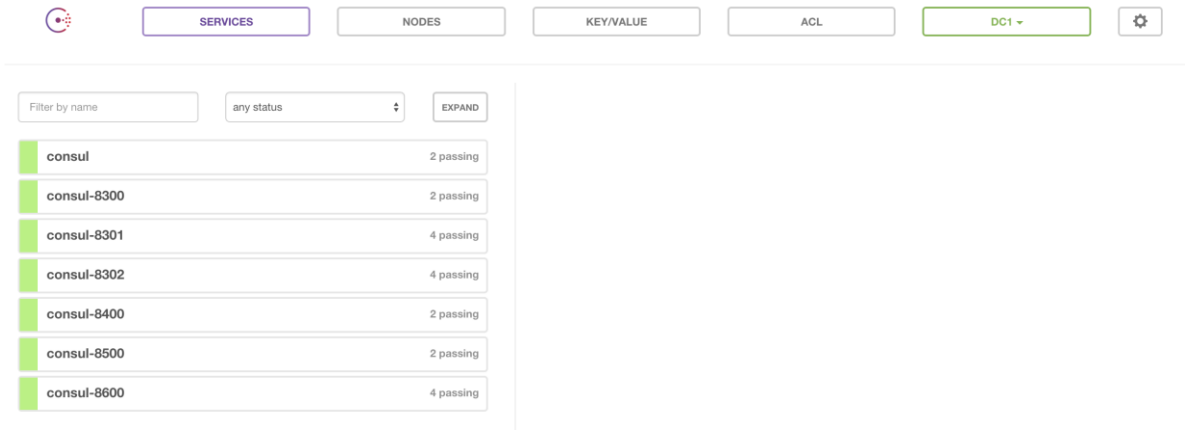
```
registrator:
    image: <%= @reg_docker_image %>
    restart: always
    net: "<%= @reg_net %>"
    volumes:
      - <%= @reg_volume %>
    command: "<%= @reg_command %>"
```

```
node 'node-01' {

}

node 'node-02' {
  include consul

}

node 'node-03' {
  include consul

  }

node 'node-04' {
  include consul

  }
```

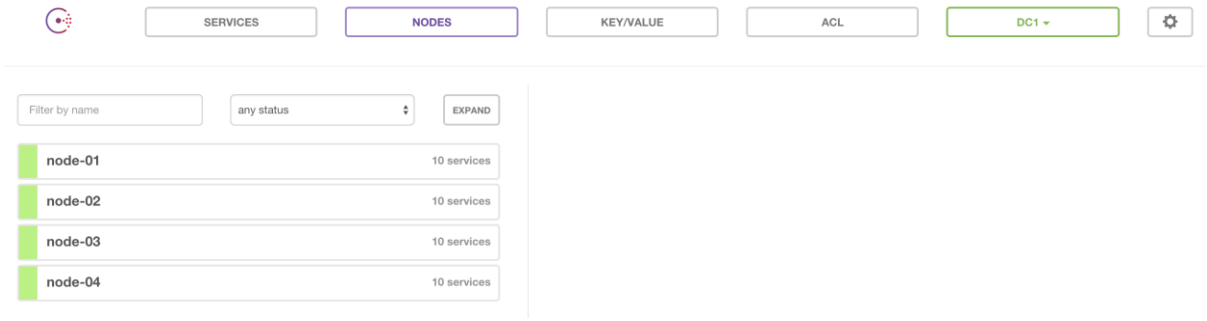

SERVICES
NODES
KEY/VALUE
ACL
DC1
Filter by name
any status
EXPAND
node-01
10 services


SERVICES
NODES
KEY/VALUE
ACL
DC1
Filter by name
any status
EXPAND
consul
1 passing
consul-8300
1 passing
consul-8301
2 passing
consul-8302
2 passing
consul-8400
1 passing
consul-8500
1 passing
consul-8600
2 passing


SERVICES
NODES
KEY/VALUE
ACL
DC1
Filter by name
any status
EXPAND
node-01
10 services
node-02
1 services




```
==> node-04: Running Puppet with default.pp...
==> node-04: Info: Loading facts
==> node-04: Info: Loading facts
==> node-04: Info: Loading facts
==> node-04: Notice: Compiled catalog for localhost in environment production in 1.15 seconds
==> node-04: Info: Applying configuration version '1457001927'
==> node-04: Notice: /Stage[main]/Docker::Repos/Yumrepo[docker]/ensure: created
==> node-04: Info: changing mode of /etc/yum.repos.d/docker.repo from 600 to 644
==> node-04: Notice: /Stage[main]/Docker::Install/Package[docker]/ensure: created
==> node-04: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]/ensure: created
==> node-04: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]: Scheduling refresh of Service[docker]
==> node-04: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d]/ensure: created
==> node-04: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]/ensure: created
==> node-04: Info: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]: Scheduling refresh of Exec[docker-systemd-reload-before-service]
==> node-04: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/returns: executed successfully
==> node-04: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]: Triggered 'refresh' from 1 events
==> node-04: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]/ensure: created
==> node-04: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]: Scheduling refresh of Service[docker]
==> node-04: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]/ensure: created
==> node-04: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]: Scheduling refresh of Service[docker]
==> node-04: Notice: /Stage[main]/Docker::Service/Service[docker]/ensure: ensure changed 'stopped' to 'running'
==> node-04: Info: /Stage[main]/Docker::Service/Service[docker]: Unscheduling refresh on Service[docker]
==> node-04: Notice: /Stage[main]/Docker::Compose/Exec[Install Docker Compose 1.6.0]/returns: executed successfully
==> node-04: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose-1.6.0]/mode: mode changed '0644' to '0755'
==> node-04: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose]/ensure: created
==> node-04: Notice: /Stage[main]/Consul::Install/File[/root/docker-compose.yml]/ensure: defined content as '{md5}95d069d208c89b0e74110e59d7084551'
==> node-04: Info: Checking if docker-compose.yml exists
==> node-04: Info: bring up containers
==> node-04: Notice: /Stage[main]/Consul::Install/Docker_compose[undef]/ensure: created
==> node-04: Notice: /Stage[main]/Consul::Package/Package[bind]/ensure: created
==> node-04: Info: Computing checksum on file /etc/named.conf
==> node-04: Info: /Stage[main]/Consul::Package/File[/etc/named.conf]: Filebucketed /etc/named.conf to puppet with sum 0d049950431ec70b91d3b56cf33bb383
==> node-04: Notice: /Stage[main]/Consul::Package/File[/etc/named.conf]/content: content changed '{md5}0d049950431ec70b91d3b56cf33bb383' to '{md5}15fa26b437cf7587b8dd7d53bce7d788'
==> node-04: Notice: /Stage[main]/Consul::Package/File[/etc/named.conf]/group: group changed 'named' to 'root'
==> node-04: Notice: /Stage[main]/Consul::Package/File[/etc/named.conf]/mode: mode changed '0640' to '0644'
==> node-04: Notice: /Stage[main]/Consul::Package/File[/etc/named/consul.conf]/ensure: created
==> node-04: Info: /Stage[main]/Consul::Package/File[/etc/named/consul.conf]: Scheduling refresh of Service[named]
==> node-04: Notice: /Stage[main]/Consul::Package/Service[named]/ensure: ensure changed 'stopped' to 'running'
==> node-04: Info: /Stage[main]/Consul::Package/Service[named]: Unscheduling refresh on Service[named]
==> node-04: Info: checking if docker network exists
==> node-04: Notice: Finished catalog run in 119.61 seconds
```




```
[vagrant@node-03 ~]$ ping consul-8500.service.consul
PING consul-8500.service.consul (172.17.8.103) 56(84) bytes of data.
64 bytes from node-03 (172.17.8.103): icmp_seq=1 ttl=64 time=0.020 ms
64 bytes from node-03 (172.17.8.103): icmp_seq=2 ttl=64 time=0.054 ms
64 bytes from node-03 (172.17.8.103): icmp_seq=3 ttl=64 time=0.047 ms
64 bytes from node-03 (172.17.8.103): icmp_seq=4 ttl=64 time=0.047 ms
64 bytes from node-03 (172.17.8.103): icmp_seq=5 ttl=64 time=0.049 ms
64 bytes from node-03 (172.17.8.103): icmp_seq=6 ttl=64 time=0.048 ms
64 bytes from node-03 (172.17.8.103): icmp_seq=7 ttl=64 time=0.047 ms
64 bytes from node-03 (172.17.8.103): icmp_seq=8 ttl=64 time=0.048 ms
64 bytes from node-03 (172.17.8.103): icmp_seq=9 ttl=64 time=0.039 ms
64 bytes from node-03 (172.17.8.103): icmp_seq=10 ttl=64 time=0.114 ms
64 bytes from node-03 (172.17.8.103): icmp_seq=11 ttl=64 time=0.047 ms
64 bytes from node-03 (172.17.8.103): icmp_seq=12 ttl=64 time=0.046 ms
^C
--- consul-8500.service.consul ping statistics ---
12 packets transmitted, 12 received, 0% packet loss, time 11005ms
rtt min/avg/max/mdev = 0.020/0.050/0.114/0.021 ms
[vagrant@node-03 ~]$
```

```
[root@node-03 ~]# docker network ls
NETWORK ID          NAME                DRIVER
b43a47a70589        docker-internal     overlay
ce2f0f870d53        none                null
290dbf24b3f6        host                host
2c0dc5e17f19        bridge              bridge
[root@node-03 ~]#
```

```
class elasticsearch {

  docker::image { 'elasticsearch:2.1.0': } ->

  docker::run { 'elasticsearch':
    image      => 'elasticsearch:2.1.0',
    net        => 'docker-internal',
    command    => 'elasticsearch -Des.network.host=0.0.0.0',
    volumes    => ['/root/esdata:/usr/share/elasticsearch/data'],
    privileged => true,
  }
}
```

```
node 'node-02' {
  include consul
  contain elasticsearch
}
```

```
class logstash {


  file { '/root/logstash-config':
    ensure => directory,
  } ->

  file { '/root/logstash-config/logstash.conf':
    ensure => file,
    content => template("logstash/logstash.conf.erb"),
  } ->

  docker::image { 'logstash:2.1.0': } ->

  docker::run { 'logstash':
    image   => 'logstash:2.1.0',
    net     => 'docker-internal',
    volumes => ['/root/logstash-config:/opt/logstash/conf.d/'],
    ports   => ['9998:9998', '9999:9999/udp', '5000:5000', '5000:5000/udp'],
    env     => ['ES_HOST=elasticsearch', 'ES_PORT=9200'],
    command => 'logstash -f /opt/logstash/conf.d/logstash.conf --debug',
  }
}
```

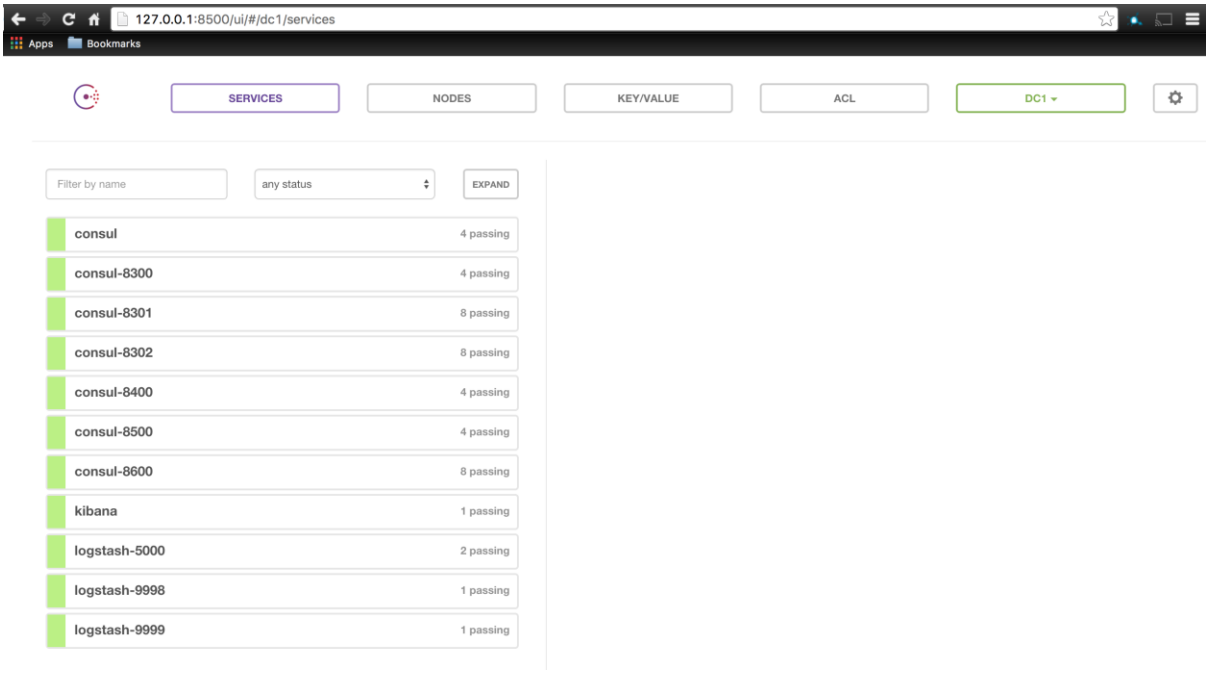
```
input {
  tcp {
    port => 5000
    type => syslog
  }
  udp {
    port => 5000
    type => syslog
  }

   file {
    type => "syslog"
    path => [ "/var/log/*.log", "/var/log/messages", "/var/log/syslog" ]
    start_position => "beginning"
  }

  file {
    type => "logstash"
    path => [ "/var/log/logstash/logstash.log" ]
    start_position => "beginning"
    }
  }

  filter {
    if [type] == "docker" {
      json {
        source => "message"
      }
    mutate {
      rename => [ "log", "message" ]
    }
    date {
      match => [ "time", "ISO8601" ]
    }
  }

 if [type] == "syslog" {
    grok {
      match => { "message" => "%{SYSLOG5424PRI}%{NONNEGINT:ver} +(?:%{TIMESTAMP_ISO8601:ts}|-) +(?:%{HOSTNAME:containerid}|-) +(?:%{NOTSPACE:containername}|-) +(?:%{NOTSPACE:proc}|-) +(?:%{WORD:msgid}|-) +(
      ?:%{SYSLOG5424SD:sd}|-|) +%{GREEDYDATA:msg}" }
    }

    syslog_pri { }
    date {
      match => [ "syslog_timestamp", "MMM  d HH:mm:ss", "MMM dd HH:mm:ss" ]
    }
    if !("_grokparsefailure" in [tags]) {
      mutate {
        replace => [ "@source_host", "%{syslog_hostname}" ]
        replace => [ "@message", "%{syslog_message}" ]
      }
    }
    mutate {
      remove_field => [ "syslog_hostname", "syslog_message", "syslog_timestamp" ]
    }
  }
}

output {
  elasticsearch { hosts => ["elasticsearch:9200"] }

  stdout { codec => rubydebug }
}
```

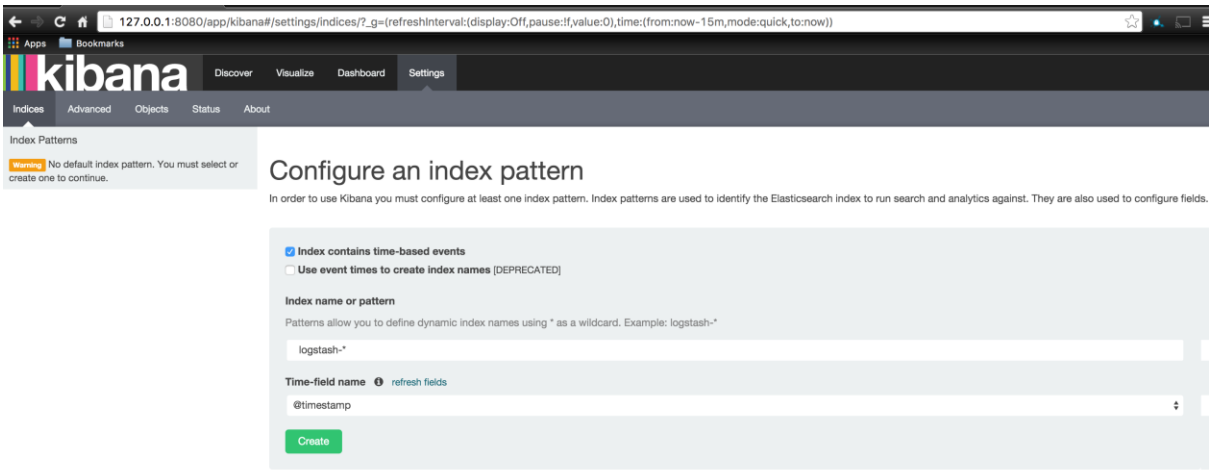
```
node 'node-03' {
  include consul
  contain logstash
  }
```

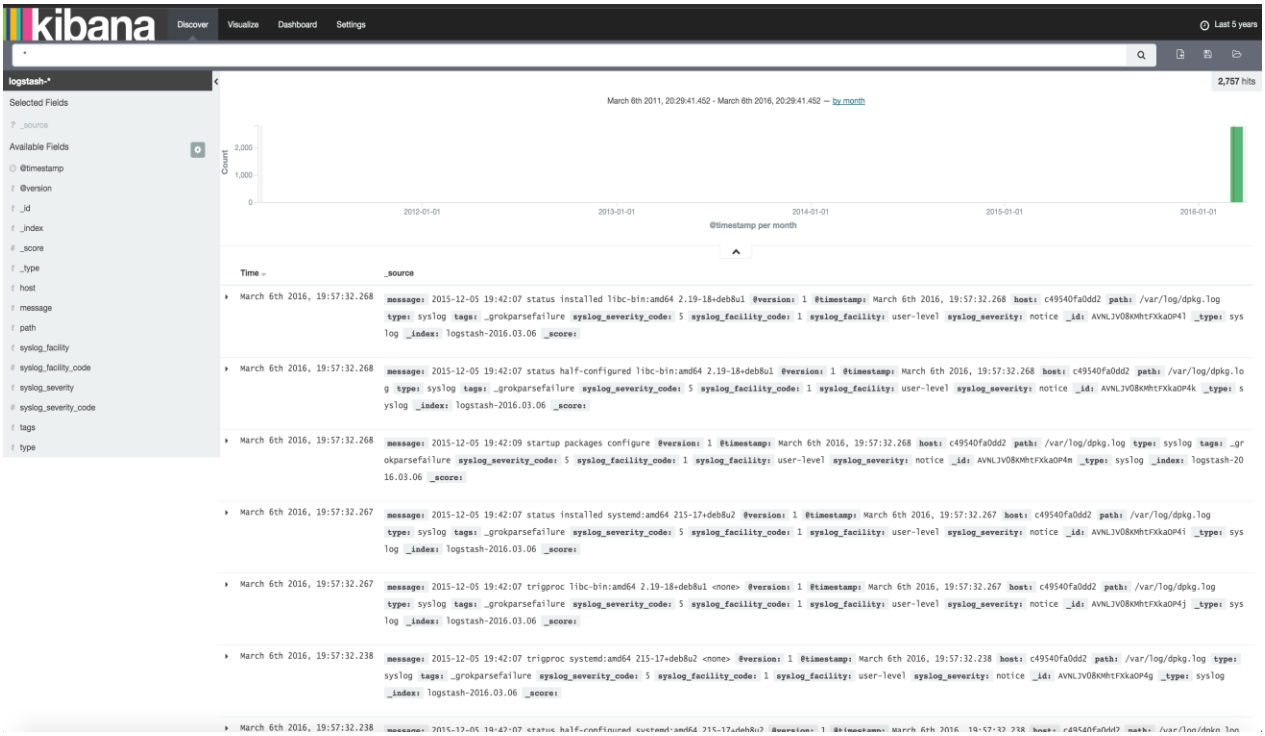
```
class kibana {

  docker::image { 'kibana:4.3.0': } ->

  docker::run { 'kibana':
    image   => 'kibana:4.3.0',
    net     => 'docker-internal',
    ports   => ['80:5601'],
    env     => ['ELASTICSEARCH_URL=http://elasticsearch:9200']
    }
 }
```

```
  node 'node-01' {
    include consul
    }

  node 'node-02' {
    include consul
    contain elasticsearch
  }

  node 'node-03' {
    include consul
    contain logstash
    }

  node 'node-04' {
    include consul
    contain kibana
    }
```

```
==> node-04: Running Puppet with default.pp...
==> node-04: Info: Loading facts
==> node-04: Info: Loading facts
==> node-04: Info: Loading facts
==> node-04: Notice: Compiled catalog for localhost in environment production in 1.28 seconds
==> node-04: Info: Applying configuration version '1457146461'
==> node-04: Notice: /Stage[main]/Docker::Repos/Yumrepo[docker]/ensure: created
==> node-04: Info: changing mode of /etc/yum.repos.d/docker.repo from 600 to 644
==> node-04: Notice: /Stage[main]/Docker::Install/Package[docker]/ensure: created
==> node-04: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]/ensure: created
==> node-04: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]: Scheduling refresh of Service[docker]
==> node-04: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d]/ensure: created
==> node-04: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]/ensure: created
==> node-04: Info: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]: Scheduling refresh of Exec[docker-systemd-reload-before-service]
==> node-04: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/returns: executed successfully
==> node-04: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]: Triggered 'refresh' from 1 events
==> node-04: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]/ensure: created
==> node-04: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]: Scheduling refresh of Service[docker]
==> node-04: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]/ensure: created
==> node-04: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]: Scheduling refresh of Service[docker]
==> node-04: Notice: /Stage[main]/Docker::Service/Service[docker]/ensure: ensure changed 'stopped' to 'running'
==> node-04: Info: /Stage[main]/Docker::Service/Service[docker]: Unscheduling refresh on Service[docker]
==> node-04: Notice: /Stage[main]/Docker::Compose/Exec[Install Docker Compose 1.6.0]/returns: executed successfully
==> node-04: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose-1.6.0]/mode: mode changed '0644' to '0755'
==> node-04: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose]/ensure: created
==> node-04: Notice: /Stage[main]/Consul::Install/File[/root/docker-compose.yml]/ensure: defined content as '{md5}95d069d208c89b0e74110e59d7084551'
==> node-04: Info: Checking if docker-compose.yml exists
==> node-04: Info: bring up containers
==> node-04: Notice: /Stage[main]/Consul::Install/Docker_compose[undef]/ensure: created
==> node-04: Notice: /Stage[main]/Consul::Package/Package[bind]/ensure: created
==> node-04: Info: Computing checksum on file /etc/named.conf
==> node-04: Info: /Stage[main]/Consul::Package/File[/etc/named.conf]: Filebucketed /etc/named.conf to puppet with sum 0d049950431ec70b91d3b56cf33bb383
==> node-04: Notice: /Stage[main]/Consul::Package/File[/etc/named.conf]/content: content changed '{md5}0d049950431ec70b91d3b56cf33bb383' to '{md5}15fa26b437cf7587b8dd7d53bce7d788'
==> node-04: Notice: /Stage[main]/Consul::Package/File[/etc/named.conf]/group: group changed 'named' to 'root'
==> node-04: Notice: /Stage[main]/Consul::Package/File[/etc/named.conf]/mode: mode changed '0640' to '0644'
==> node-04: Notice: /Stage[main]/Consul::Package/File[/etc/named/consul.conf]/ensure: created
==> node-04: Info: /Stage[main]/Consul::Package/File[/etc/named/consul.conf]: Scheduling refresh of Service[named]
==> node-04: Notice: /Stage[main]/Consul::Package/Service[named]/ensure: ensure changed 'stopped' to 'running'
==> node-04: Info: /Stage[main]/Consul::Package/Service[named]: Unscheduling refresh on Service[named]
==> node-04: Info: checking if docker network exists
==> node-04: Notice: /Stage[main]/Kibana/Docker::Image[kibana]/File[/usr/local/bin/update_docker_image.sh]/ensure: created
==> node-04: Notice: /Stage[main]/Kibana/Docker::Image[kibana]/Exec[/usr/local/bin/update_docker_image.sh kibana]/returns: executed successfully
==> node-04: Notice: /Stage[main]/Kibana/Docker::Run[kibana]/File[/etc/systemd/system/docker-kibana.service]/ensure: created
==> node-04: Info: /Stage[main]/Kibana/Docker::Run[kibana]/File[/etc/systemd/system/docker-kibana.service]: Scheduling refresh of Exec[docker-systemd-reload]
==> node-04: Info: /Stage[main]/Kibana/Docker::Run[kibana]/File[/etc/systemd/system/docker-kibana.service]: Scheduling refresh of Service[docker-kibana]
==> node-04: Notice: /Stage[main]/Docker::Systemd_reload/Exec[docker-systemd-reload]: Triggered 'refresh' from 1 events
==> node-04: Notice: /Stage[main]/Kibana/Docker::Run[kibana]/Service[docker-kibana]/ensure: ensure changed 'stopped' to 'running'
==> node-04: Info: /Stage[main]/Kibana/Docker::Run[kibana]/Service[docker-kibana]: Unscheduling refresh on Service[docker-kibana]
==> node-04: Notice: Finished catalog run in 114.39 seconds
```


127.0.0.1:8500/ui/#/dc1/services
Apps
Bookmarks
SERVICES
NODES
KEY/VALUE
ACL
DC1
Filter by name
any status
EXPAND
consul 4 passing
consul-8300 4 passing
consul-8301 8 passing
consul-8302 8 passing
consul-8400 4 passing
consul-8500 4 passing
consul-8600 8 passing
kibana 1 passing
logstash-5000 2 passing
logstash-9998 1 passing
logstash-9999 1 passing


127.0.0.1:8080/app/kibana#/settings/indices/?_g=(refreshInterval:(display:Off,pause:!f,value:0),time:(from:now-15m,mode:quick,to:now))
Apps
Bookmarks
kibana
Discover
Visualize
Dashboard
Settings
Indices
Advanced
Objects
Status
About
Index Patterns
Warning No default index pattern. You must select or create one to continue.
Configure an index pattern
In order to use Kibana you must configure at least one index pattern. Index patterns are used to identify the Elasticsearch index to run search and analytics against. They are also used to configure fields.
Index contains time-based events
Use event times to create index names [DEPRECATED]
Index name or pattern
Patterns allow you to define dynamic index names using * as a wildcard. Example: logstash-*
logstash-*
Time-field name
refresh fields
@timestamp
Create


kibana
Discover
Visualize
Dashboard
Settings
Last 5 years
logstash-*
Selected Fields
_source
Available Fields
@timestamp
@version
_id
_index
_score
_type
host
message
path
syslog_facility
syslog_facility_code
syslog_severity
syslog_severity_code
tags
type
2,757 hits
March 6th 2011, 20:29:41.452 - March 6th 2016, 20:29:41.452 — by month
Count
2,000
1,000
0
2012-01-01
2013-01-01
2014-01-01
2015-01-01
2016-01-01
@timestamp per month
Time
_source
March 6th 2016, 19:57:32.268
message: 2015-12-05 19:42:07 status installed libc-bin:amd64 2.19-18+deb8u1 @version: 1 @timestamp: March 6th 2016, 19:57:32.268 host: c49540fa0dd2 path: /var/log/dpkg.log type: syslog tags: _grokparsefailure syslog_severity_code: 5 syslog_facility_code: 1 syslog_facility: user-level syslog_severity: notice _id: AVNLJVO8KMhtFXkaOP41 _type: syslog _index: logstash-2016.03.06 _score:
March 6th 2016, 19:57:32.268
message: 2015-12-05 19:42:07 status half-configured libc-bin:amd64 2.19-18+deb8u1 @version: 1 @timestamp: March 6th 2016, 19:57:32.268 host: c49540fa0dd2 path: /var/log/dpkg.log type: syslog tags: _grokparsefailure syslog_severity_code: 5 syslog_facility_code: 1 syslog_facility: user-level syslog_severity: notice _id: AVNLJVO8KMhtFXkaOP4k _type: syslog _index: logstash-2016.03.06 _score:
March 6th 2016, 19:57:32.268
message: 2015-12-05 19:42:09 startup packages configure @version: 1 @timestamp: March 6th 2016, 19:57:32.268 host: c49540fa0dd2 path: /var/log/dpkg.log type: syslog tags: _grokparsefailure syslog_severity_code: 5 syslog_facility_code: 1 syslog_facility: user-level syslog_severity: notice _id: AVNLJVO8KMhtFXkaOP4m _type: syslog _index: logstash-2016.03.06 _score:
March 6th 2016, 19:57:32.267
message: 2015-12-05 19:42:07 status installed systemd:amd64 215-17+deb8u2 @version: 1 @timestamp: March 6th 2016, 19:57:32.267 host: c49540fa0dd2 path: /var/log/dpkg.log type: syslog tags: _grokparsefailure syslog_severity_code: 5 syslog_facility_code: 1 syslog_facility: user-level syslog_severity: notice _id: AVNLJVO8KMhtFXkaOP4i _type: syslog _index: logstash-2016.03.06 _score:
March 6th 2016, 19:57:32.267
message: 2015-12-05 19:42:07 trigproc libc-bin:amd64 2.19-18+deb8u1 <none> @version: 1 @timestamp: March 6th 2016, 19:57:32.267 host: c49540fa0dd2 path: /var/log/dpkg.log type: syslog tags: _grokparsefailure syslog_severity_code: 5 syslog_facility_code: 1 syslog_facility: user-level syslog_severity: notice _id: AVNLJVO8KMhtFXkaOP4j _type: syslog _index: logstash-2016.03.06 _score:
March 6th 2016, 19:57:32.238
message: 2015-12-05 19:42:07 trigproc systemd:amd64 215-17+deb8u2 <none> @version: 1 @timestamp: March 6th 2016, 19:57:32.238 host: c49540fa0dd2 path: /var/log/dpkg.log type: syslog tags: _grokparsefailure syslog_severity_code: 5 syslog_facility_code: 1 syslog_facility: user-level syslog_severity: notice _id: AVNLJVO8KMhtFXkaOP4g _type: syslog _index: logstash-2016.03.06 _score:
March 6th 2016, 19:57:32.238

Chapter 7: Container Schedulers

```
#!/usr/bin/ruby env

require "socket"
$hostname = Socket.gethostname

forge 'http://forge.puppetlabs.com'


mod 'puppetlabs/stdlib'
mod 'puppetlabs/vcsrepo'
mod 'nanliu/staging'
mod 'KyleAnderson/consul'
mod 'scottyc/docker_swarm'
mod 'scottyc/golang'
mod 'garethr/docker', :git => "https://github.com/scotty-c/garethr-docker.git"
mod 'stankevich/python'
mod 'stahnma/epel'
mod 'maestrodev/wget'
```

```
-
  box: scottyc/centos-7-puppet-kernel-4-4
  cpu: 1
  ip: "172.17.8.101"
  name: swarm-101
  forward_ports:
      - { guest: 8500, host: 9501 }
      - { guest: 80, host: 8001 }
      - { guest: 443, host: 8441 }
      - { guest: 8080, host: 8081 }
  ram: 4096
  shell_commands:
      - { shell: yum install -y git wget curl lvm2 unzip device-mapper-libs && systemctl stop firewalld && systemctl disable firewalld }
      - { shell: 'echo -e "PEERDNS=no\nDNS1=127.0.0.1\nDNS2=8.8.8.8">>/etc/sysconfig/network-scripts/ifcfg-enp0s3 && systemctl restart network'}
      - { shell: /opt/puppet/bin/gem install r10k }
      - { shell: 'echo -e "172.17.8.101 swarm-101">/etc/hosts && echo "PATH=\$PATH:/usr/local/bin" >> ~/.bashrc' }
      - { shell: cp /home/vagrant/swarm-101/Puppetfile /tmp && cd /tmp && /opt/puppet/bin/r10k puppetfile install -v }
      - { shell: cp /home/vagrant/swarm-101/modules/* -R /tmp/modules }
```

```
-
  box: scottyc/centos-7-puppet-kernel-4-4
  cpu: 1
  ip: "172.17.8.102"
  name: swarm-102
  forward_ports:
      - { guest: 8500, host: 9502 }
      - { guest: 80, host: 8002 }
      - { guest: 443, host: 8442 }
      - { guest: 8080, host: 8082 }
  ram: 4096
  shell_commands:
      - { shell: yum install -y git wget curl lvm2 device-mapper-libs unzip && systemctl stop firewalld && systemctl disable firewalld  }
      - { shell: 'echo -e "PEERDNS=no\nDNS1=127.0.0.1\nDNS2=8.8.8.8">>/etc/sysconfig/network-scripts/ifcfg-enp0s3 && systemctl restart network'}
      - { shell: /opt/puppet/bin/gem install r10k }
      - { shell: 'echo -e "172.17.8.101 swarm-101\n172.17.8.102 swarm-102">/etc/hosts && echo "PATH=\$PATH:/usr/local/bin" >> ~/.bashrc' }
      - { shell: cp /home/vagrant/swarm-102/Puppetfile /tmp && cd /tmp && /opt/puppet/bin/r10k puppetfile install -v }
      - { shell: cp /home/vagrant/swarm-102/modules/* -R /tmp/modules }
```

```
-
  box: scottyc/centos-7-puppet-kernel-4-4
  cpu: 1
  ip: "172.17.8.103"
  name: swarm-103
  forward_ports:
      - { guest: 8500, host: 9503 }
      - { guest: 80, host: 8003 }
      - { guest: 443, host: 8443 }
      - { guest: 8080, host: 8083 }
  ram: 4096
  shell_commands:
      - { shell: yum install -y git wget curl lvm2 unzip device-mapper-libs && systemctl stop firewalld && systemctl disable firewalld  }
      - { shell: 'echo -e "PEERDNS=no\nDNS1=127.0.0.1\nDNS2=8.8.8.8">>/etc/sysconfig/network-scripts/ifcfg-enp0s3 && systemctl restart network'}
      - { shell: /opt/puppet/bin/gem install r10k }
      - { shell: 'echo -e "172.17.8.101 swarm-101\n172.17.8.103 swarm-103">/etc/hosts && echo "PATH=\$PATH:/usr/local/bin" >> ~/.bashrc' }
      - { shell: cp /home/vagrant/swarm-103/Puppetfile /tmp && cd /tmp && /opt/puppet/bin/r10k puppetfile install -v }
      - { shell: cp /home/vagrant/swarm-103/modules/* -R /tmp/modules }
```

```
-
  box: scottyc/centos-7-puppet-kernel-4-4
  cpu: 1
  ip: "172.17.8.114"
  name: swarm-master-01
  forward_ports:
      - { guest: 8500, host: 9504 }
  ram: 2048
  shell_commands:
      - { shell: yum install -y git wget curl lvm2 unzip device-mapper-libs && systemctl stop firewalld && systemctl disable firewalld  }
      - { shell: 'echo -e "PEERDNS=no\nDNS1=127.0.0.1\nDNS2=8.8.8.8">>/etc/sysconfig/network-scripts/ifcfg-enp0s3 && systemctl restart network'}
      - { shell: /opt/puppet/bin/gem install r10k }
      - { shell: 'echo -e "172.17.8.101 swarm-101\n172.17.8.114 swarm-master-01">/etc/hosts && echo "PATH=\$PATH:/usr/local/bin" >> ~/.bashrc' }
      - { shell: cp /home/vagrant/swarm-master-01/Puppetfile /tmp && cd /tmp && /opt/puppet/bin/r10k puppetfile install -v }
      - { shell: cp /home/vagrant/swarm-master-01/modules/* -R /tmp/modules }
```

```
-
  box: puppetlabs/centos-7.0-64-puppet-enterprise
  cpu: 1
  ip: "172.17.8.115"
  name: swarm-master-02
  forward_ports:
      - { guest: 8500, host: 9505 }
  ram: 2048
  shell_commands:
      - { shell: yum install -y git wget curl lvm2 unzip device-mapper-libs && systemctl stop firewalld && systemctl disable firewalld  }
      - { shell: 'echo -e "PEERDNS=no\nDNS1=127.0.0.1\nDNS2=8.8.8.8">>/etc/sysconfig/network-scripts/ifcfg-enp0s3 && systemctl restart network'}
      - { shell: /opt/puppet/bin/gem install r10k }
      - { shell: 'echo -e "172.17.8.101 swarm-101\n172.17.8.115 swarm-master-02">/etc/hosts && echo "PATH=\$PATH:/usr/local/bin" >> ~/.bashrc' }
      - { shell: cp /home/vagrant/swarm-master-02/Puppetfile /tmp && cd /tmp && /opt/puppet/bin/r10k puppetfile install -v }
      - { shell: cp /home/vagrant/swarm-master-02/modules/* -R /tmp/modules }
```

```
:backends:
  - yaml
:hierarchy:
  - global

:yaml:
  :datadir: /vagrant/hieradata
```

```
docker_swarm::swarm_version: v1.1.3
consul::version: 0.6.3
```

```
# == Class: config
#
#
class config(

  $consul_ip = "$::ipaddress_enp0s8",

){

  include config::consul_config
  contain config::swarm
  contain config::compose
  contain config::dns
  contain config::run_containers

  Class['config::swarm'] -> Class['config::compose'] -> Class['config::dns'] -> Class['config::run_containers']
}


```


class config::compose {
if $hostname =~ /^swarm-master*/ {
notice ["This server is the Swarm Manager."]
}
else {
file { '/root/docker-compose.yml':
ensure => file,
content => template("config/registrator.yml.erb"),
} ->
docker_compose {'swarm app':
ensure => present,
source => '/root',
scale => ['1']
}
}
}

```
class config::consul_config {

if $hostname =~ /^*101*$/ {

  class { 'consul':
    config_hash => {
      'datacenter'       => 'dev',
      'data_dir'         => '/opt/consul',
      'ui_dir'           => '/opt/consul/ui',
      'bind_addr'        => $::ipaddress_enp0s8,
      'client_addr'      => '0.0.0.0',
      'node_name'        => "$::hostname",
      'advertise_addr'   => '172.17.8.101',
      'bootstrap_expect' => '1',
      'server'           => true
       }
    }
  }

else {

 class { 'consul':
  config_hash => {
      'bootstrap'      => false,
      'datacenter'     => 'dev',
      'data_dir'       => '/opt/consul',
      'ui_dir'         => '/opt/consul/ui',
      'bind_addr'      => $::ipaddress_enp0s8,
      'client_addr'    => '0.0.0.0',
      'node_name'      => "$::hostname",
      'advertise_addr' => $::ipaddress_enp0s8,
      'start_join'     => ['172.17.8.101','172.17.8.103','172.17.8.103'],
      'server'         => false
      }
    }
  }

  ::consul::service { 'docker-service':
  checks  => [
    {
    script   => 'service docker status',
    interval => '10s',
    tags     => ['docker-service']
    }
  ],
  address => $::ipaddress_enp0s8,
  }
}
```

```
class config::dns {

  package { 'bind':
    ensure => present
  } ->

  file { '/etc/named.conf':
    ensure  => present,
    content => template("config/named.conf.erb"),
    mode    => '0644',
    owner   => 'root',
    group   => 'root',
    require => Package['bind'],
  } ~>

  file { '/etc/named/consul.conf':
    ensure  => present,
    content => template("config/consul.conf.erb"),
    mode    => '0644',
    owner   => 'root',
    group   => 'root',
    require => Package['bind'],
  } ~>


  service { 'named':
    ensure  => running,
    enable  => true,
    require => File['/etc/named.conf'],
  }
}
```

```
class config::run_containers {

  if $hostname =~ /swarm-master-02/ {

    swarm_run {'jenkins':
    ensure  => present,
    image   => 'jenkins',
    ports   => ['8080:8080'],
    require => Class['config::swarm']
    }

    swarm_run {'nginx':
    ensure     => present,
    image      => 'nginx',
    ports      => ['80:80', '443:443'],
    log_driver => 'syslog',
    network    => 'swarm-private',
    require    => Class['config::swarm']
     }

    swarm_run {'redis':
    ensure  => present,
    image   => 'redis',
    network => 'swarm-private',
    require => Class['config::swarm']
     }
   }
 }
```

```
class config::swarm {

  class { 'docker_swarm':
  require => Class['config::consul_config']
  }

  docker_network { 'swarm-private':
  ensure => present,
  create => true,
  driver => 'overlay',
  require => Class['config::consul_config']
  }

  if $hostname =~ /^swarm-master*/ {

  swarm_cluster {'cluster 1':
    ensure       => present,
    backend      => 'consul',
    cluster_type => 'manage',
    port         => '8500',
    address      => '172.17.8.101',
    advertise    => $::ipaddress_enp0s8,
    path         => 'swarm',
    }
  }

  else {

  swarm_cluster {'cluster 1':
   ensure       => present,
   backend      => 'consul',
   cluster_type => 'join',
   port         => '8500',
   address      => '172.17.8.101',
   path         => 'swarm'
    }
  }
}
```

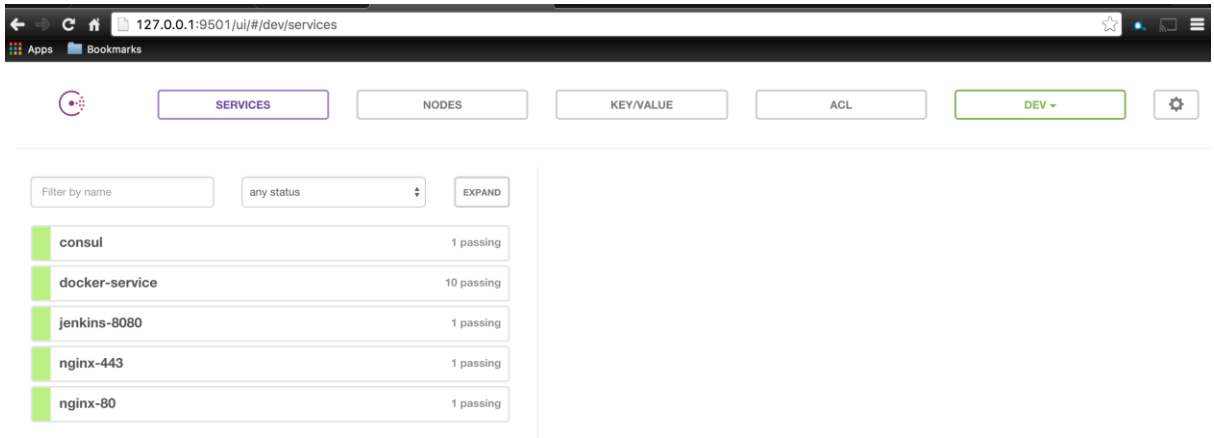
```
zone "consul" IN {
  type forward;
  forward only;
  forwarders { 127.0.0.1 port 8600; };
};
```

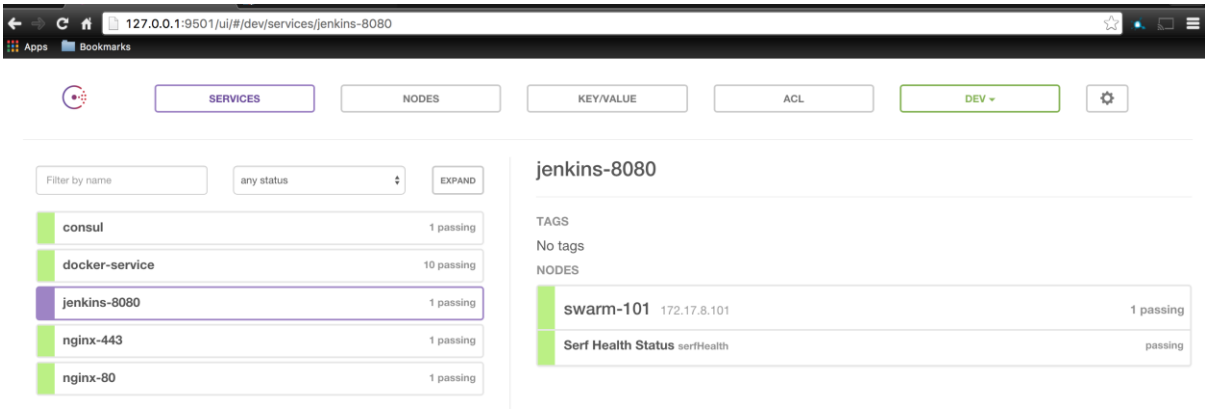
```
options {
  listen-on port 53 { 127.0.0.1; };
  listen-on-v6 port 53 { ::1; };
  directory       "/var/named";
  dump-file       "/var/named/data/cache_dump.db";
  statistics-file "/var/named/data/named_stats.txt";
  memstatistics-file "/var/named/data/named_mem_stats.txt";
  allow-query     { localhost; };
  recursion yes;

  dnssec-enable no;
  dnssec-validation no;

  /* Path to ISC DLV key */
  bindkeys-file "/etc/named.iscdlv.key";

  managed-keys-directory "/var/named/dynamic";
};

include "/etc/named/consul.conf";
```

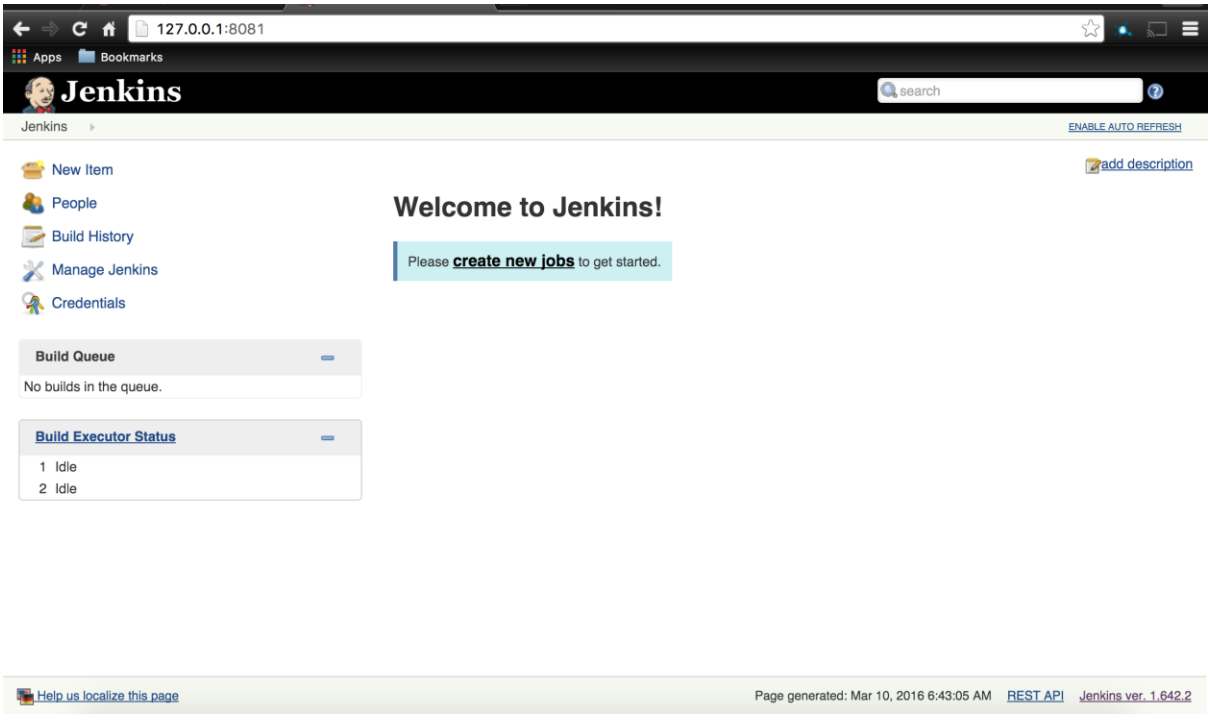
```
registrator:
  image: gliderlabs/registrator
  net: "host"
  volumes:
   - "/var/run/docker.sock:/tmp/docker.sock"
  command: consul://<%= @consul_ip %>:8500
```

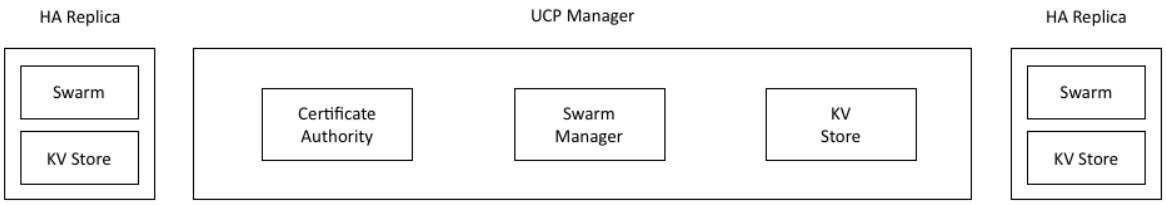
```
1   node 'swarm-101' { include config }
2
3   node 'swarm-102' { include config }
4
5   node 'swarm-103' { include config }
6
7   node 'swarm-master-01' { include config }
8
9   node 'swarm-master-02' { include config }
10
```

```
==> swarm-master-02: Debug: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/unless:
==> swarm-master-02: Debug: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/unless: Mar 10 04:20:33 swarm-master-02 docker[5881]: time="2016-03-10T04:20:33.096553554Z" level=info msg="2016/03/10 04:20:33 [INFO] memberlist: Suspect swarm-master-01 h
as failed, no acks received\n"
==> swarm-master-02: Debug: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/unless: Mar 10 04:20:33 swarm-master-02 docker[5881]: time="2016-03-10T04:20:33.641443075Z" level=info msg="2016/03/10 04:20:33 [INFO] memberlist: Marking swarm-master-01 a
s failed, suspect timeout reached\n"
==> swarm-master-02: Debug: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/unless: Mar 10 04:20:33 swarm-master-02 docker[5881]: time="2016-03-10T04:20:33.641545156Z" level=info msg="2016/03/10 04:20:33 [INFO] serf: EventMemberFailed: swarm-master
-01 172.17.8.114\n"
==> swarm-master-02: Debug: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/unless: Mar 10 04:20:33 swarm-master-02 docker[5881]: time="2016-03-10T04:20:33.915168793Z" level=info msg="2016/03/10 04:20:33 [INFO] serf: EventMemberJoin: swarm-master-0
1 172.17.8.114\n"
==> swarm-master-02: Debug: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/unless: Mar 10 04:21:05 swarm-master-02 docker[5881]: time="2016-03-10T04:21:05.095838722Z" level=error msg="2016/03/10 04:21:05 [ERR] memberlist: Failed to send ping: writ
e udp 172.17.8.115:7946->172.17.8.114:7946: sendto: invalid argument\n"
==> swarm-master-02: Debug: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/unless: Mar 10 04:21:06 swarm-master-02 docker[5881]: time="2016-03-10T04:21:06.097012030Z" level=error msg="2016/03/10 04:21:06 [ERR] memberlist: Failed to send ping: writ
e udp 172.17.8.115:7946->172.17.8.101:7946: sendto: invalid argument\n"
==> swarm-master-02: Debug: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/unless: Mar 10 04:21:07 swarm-master-02 docker[5881]: time="2016-03-10T04:21:07.096732389Z" level=error msg="2016/03/10 04:21:07 [ERR] memberlist: Failed to send ping: writ
e udp 172.17.8.115:7946->172.17.8.103:7946: sendto: invalid argument\n"
==> swarm-master-02: Debug: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/unless: Mar 10 04:21:08 swarm-master-02 docker[5881]: time="2016-03-10T04:21:08.096745839Z" level=error msg="2016/03/10 04:21:08 [ERR] memberlist: Failed to send ping: writ
e udp 172.17.8.115:7946->172.17.8.114:7946: sendto: invalid argument\n"
==> swarm-master-02: Debug: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/unless: Mar 10 04:21:09 swarm-master-02 docker[5881]: time="2016-03-10T04:21:09.095853629Z" level=error msg="2016/03/10 04:21:09 [ERR] memberlist: Failed to send ping: writ
e udp 172.17.8.115:7946->172.17.8.102:7946: sendto: invalid argument\n"
==> swarm-master-02: Debug: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/unless: Mar 10 04:21:12 swarm-master-02 docker[5881]: time="2016-03-10T04:21:12.107953862Z" level=warning msg="2016/03/10 04:21:12 [WARN] memberlist: Refuting a dead messag
e (from: swarm-102)\n"
==> swarm-master-02: Debug: Executing '/bin/systemctl is-active docker'
==> swarm-master-02: Debug: Executing '/bin/systemctl is-enabled docker'
==> swarm-master-02: Info: checking if swarm is running
==> swarm-master-02: Debug: Executing '/bin/systemctl is-active named'
==> swarm-master-02: Debug: Executing '/bin/systemctl is-enabled named'
==> swarm-master-02: Debug: Exec[wget-get golang package](provider=posix): Executing check 'test -s '/tmp/go1.6.linux-amd64.tar.gz''
==> swarm-master-02: Debug: Executing 'test -s '/tmp/go1.6.linux-amd64.tar.gz''
==> swarm-master-02: Debug: Executing '/bin/systemctl is-active consul'
==> swarm-master-02: Debug: Executing '/bin/systemctl is-enabled consul'
==> swarm-master-02: Debug: Executing '/bin/git config -l'
==> swarm-master-02: Debug: Executing '/bin/git fetch origin'
==> swarm-master-02: Debug: Executing '/bin/git fetch --tags origin'
==> swarm-master-02: Debug: Executing '/bin/git rev-parse HEAD'
==> swarm-master-02: Debug: Executing '/bin/git tag -l'
==> swarm-master-02: Debug: Executing '/bin/git rev-parse v1.1.3^0'
==> swarm-master-02: Debug: Executing '/bin/git branch -a'
==> swarm-master-02: Debug: Executing '/bin/git branch -a'
==> swarm-master-02: Debug: Executing '/bin/git checkout --force v1.1.3'
==> swarm-master-02: Debug: Executing '/bin/git branch -a'
==> swarm-master-02: Debug: Executing '/bin/git submodule update --init --recursive'
==> swarm-master-02: Notice: /Stage[main]/Docker_swarm::Install/Vcsrepo[/usr/local/go/src/github.com/docker/swarm]/revision: revision changed '0f8d82b90932741bda5425e990df433c9ee06a7' to 'v1.1.3'
==> swarm-master-02: Debug: /Stage[main]/Docker_swarm::Install/Vcsrepo[/usr/local/go/src/github.com/docker/swarm]: The container Class[Docker_swarm::Install] will propagate my refresh event
==> swarm-master-02: Debug: Class[Docker_swarm::Install]: The container Stage[main] will propagate my refresh event
==> swarm-master-02: Info: checking if docker network exists
==> swarm-master-02: Info: checking if conatiner is running
==> swarm-master-02: Notice: /Stage[main]/Config::Run_containers/Swarm_run[jenkins]/ports: defined 'ports' as '8080:8080'
==> swarm-master-02: Debug: /Stage[main]/Config::Run_containers/Swarm_run[jenkins]: The container Class[Config::Run_containers] will propagate my refresh event
==> swarm-master-02: Info: checking if conatiner is running
==> swarm-master-02: Notice: /Stage[main]/Config::Run_containers/Swarm_run[nginx]/ports: defined 'ports' as '80:80 443:443'
==> swarm-master-02: Debug: /Stage[main]/Config::Run_containers/Swarm_run[nginx]: The container Class[Config::Run_containers] will propagate my refresh event
==> swarm-master-02: Info: checking if conatiner is running
==> swarm-master-02: Notice: /Stage[main]/Config::Run_containers/Swarm_run[redis]/ports: defined 'ports' as '0'
==> swarm-master-02: Debug: /Stage[main]/Config::Run_containers/Swarm_run[redis]: The container Class[Config::Run_containers] will propagate my refresh event
==> swarm-master-02: Debug: Class[Config::Run_containers]: The container Stage[main] will propagate my refresh event
==> swarm-master-02: Debug: Class[Config::Run_containers]: The container Class[Config] will propagate my refresh event
==> swarm-master-02: Debug: Class[Config]: The container Stage[main] will propagate my refresh event
==> swarm-master-02: Debug: Finishing transaction 33901040
==> swarm-master-02: Debug: Storing state
==> swarm-master-02: Debug: Stored state in 0.14 seconds
==> swarm-master-02: Notice: Finished catalog run in 7.38 seconds
==> swarm-master-02: Debug: Using settings: adding file resource 'rrddir': 'File[/var/opt/lib/pe-puppet/rrd]{:path=>"/var/opt/lib/pe-puppet/rrd", :mode=>"750", :owner=>"pe-puppet", :group=>"pe-puppet", :ensure=>:directory, :loglevel=>:debug, :links=>:follow, :backup=>fals
e}'
==> swarm-master-02: Debug: Finishing transaction 33083240
==> swarm-master-02: Debug: Received report to process from localhost
==> swarm-master-02: Debug: Processing report from localhost with processor Puppet::Reports::Store
```




127.0.0.1:9501/ui/#/dev/services/jenkins-8080
Apps
Bookmarks
SERVICES
NODES
KEY/VALUE
ACL
DEV
Filter by name
any status
EXPAND
consul
1 passing
docker-service
10 passing
jenkins-8080
1 passing
nginx-443
1 passing
nginx-80
1 passing
jenkins-8080
TAGS
No tags
NODES
swarm-101
172.17.8.101
1 passing
Serf Health Status
serfHealth
passing


127.0.0.1:8081
Apps
Bookmarks
Jenkins
search
Jenkins
ENABLE AUTO REFRESH
New Item
People
Build History
Manage Jenkins
Credentials
add description
Welcome to Jenkins!
Please create new jobs to get started.
Build Queue
No builds in the queue.
Build Executor Status
1 Idle
2 Idle
Help us localize this page
Page generated: Mar 10, 2016 6:43:05 AM
REST API
Jenkins ver. 1.642.2




```
{
        "auths": {
                "https://index.docker.io/v1/": {
                        "auth": "xxxxxxxxxxxxxxxxxxxx",
                        "email": "your.email@mail.com"
                }
        }
}
```

```
---
-
  box: puppetlabs/ubuntu-14.04-64-puppet-enterprise
  cpu: 2
  ip: "172.17.10.101"
  name: ucp-01
  forward_ports:
      - { guest: 443, host: 8443 }
  ram: 4096
  shell_commands:
      - { shell: 'apt-get update -y' }
      - { shell: 'apt-get install -y wget git' }
      - { shell: 'mkdir ~/.docker || true && cp /vagrant/config.json ~/.docker/' }
      - { shell: 'mkdir /etc/docker/ || true && cp /vagrant/docker_subscription.lic /etc/docker/subscription.lic' }
      - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
      - { shell: 'cp /home/vagrant/ucp-01/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
      - { shell: 'cp /home/vagrant/ucp-01/modules/* -R /tmp/modules || true' }


-
  box: puppetlabs/ubuntu-14.04-64-puppet-enterprise
  cpu: 2
  ip: "172.17.10.102"
  name: ucp-02
  forward_ports:
      - { guest: 443, host: 9443 }
  ram: 4096
  shell_commands:
      - { shell: 'apt-get update -y' }
      - { shell: 'apt-get install -y wget git' }
      - { shell: 'mkdir ~/.docker || true && cp /vagrant/config.json ~/.docker/' }
      - { shell: 'mkdir /etc/docker/ || true && cp /vagrant/docker_subscription.lic /etc/docker/subscription.lic' }
      - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
      - { shell: 'cp /home/vagrant/ucp-02/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
      - { shell: 'cp /home/vagrant/ucp-02/modules/* -R /tmp/modules || true' }

-
  box: puppetlabs/ubuntu-14.04-64-puppet-enterprise
  cpu: 2
  ip: "172.17.10.103"
  name: ucp-03
  forward_ports:
      - { guest: 443, host: 10443 }
  ram: 4096
  shell_commands:
      - { shell: 'apt-get update -y' }
      - { shell: 'apt-get install -y wget git' }
      - { shell: 'mkdir ~/.docker || true && cp /vagrant/config.json ~/.docker/' }
      - { shell: 'mkdir /etc/docker/ || true && cp /vagrant/docker_subscription.lic /etc/docker/subscription.lic' }
      - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
      - { shell: 'cp /home/vagrant/ucp-03/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
      - { shell: 'cp /home/vagrant/ucp-03/modules/* -R /tmp/modules || true' }

```

```
#!/usr/bin/ruby env

require "socket"
$hostname = Socket.gethostname

forge 'http://forge.puppetlabs.com'

mod 'garethr/docker', :git => "https://github.com/scotty-c/garethr-docker.git"
mod 'puppetlabs/apt'
mod 'puppetlabs/docker_ucp'
mod 'puppetlabs/stdlib'


```

```
ucpconfig::ucp_url: https://172.17.10.101
ucpconfig::ucp_fingerprint:

```

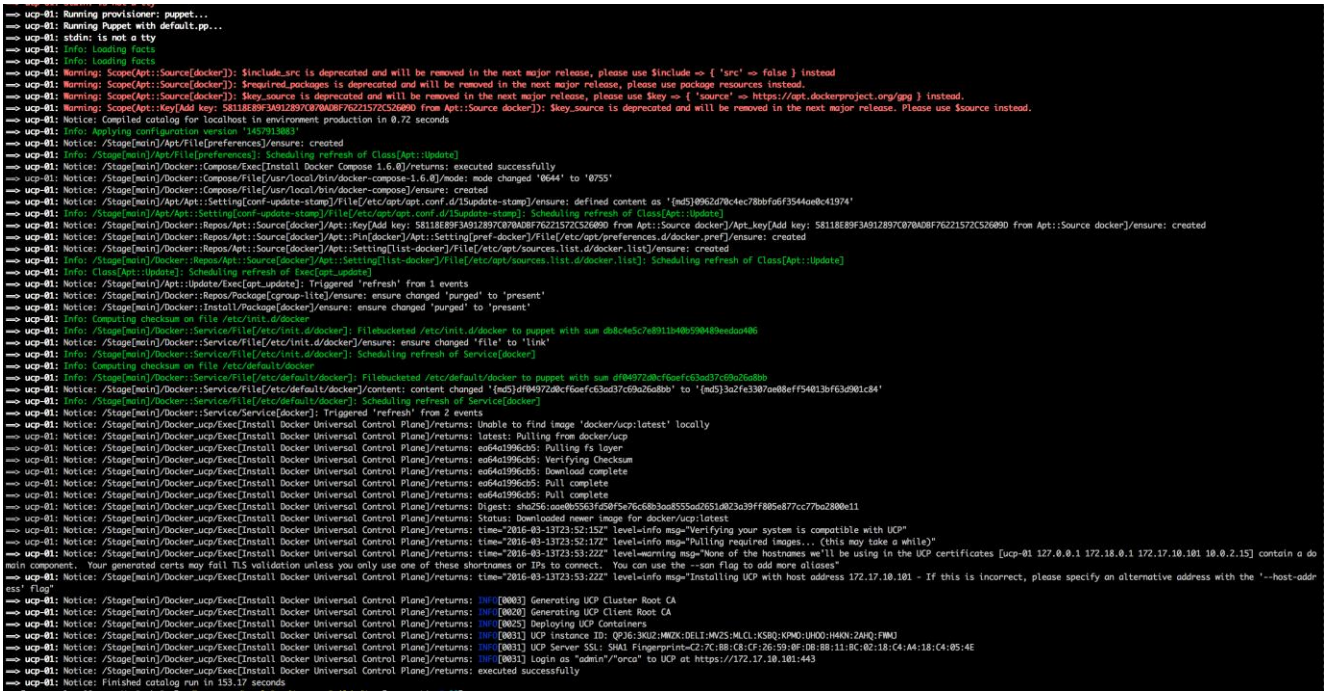
```
class ucpconfig::params {

  $ucp_url         = ''
  $ucp_username    = 'admin'
  $ucp_password    = 'orca'
  $ucp_fingerprint = ''
}
```

```
class ucpconfig (

  $ucp_url         = $ucpconfig::params::ucp_url,
  $ucp_username    = $ucpconfig::params::ucp_username,
  $ucp_password    = $ucpconfig::params::ucp_password,
  $ucp_fingerprint = $ucpconfig::params::ucp_fingerprint,
) inherits ucpconfig::params {

class { 'docker':
  socket_bind => 'unix:///var/run/docker.sock',
}

case $::hostname {
  'ucp-01': {
    include ucpconfig::master
  }
  default: {
  include ucpconfig::node
    }
  }
}
```

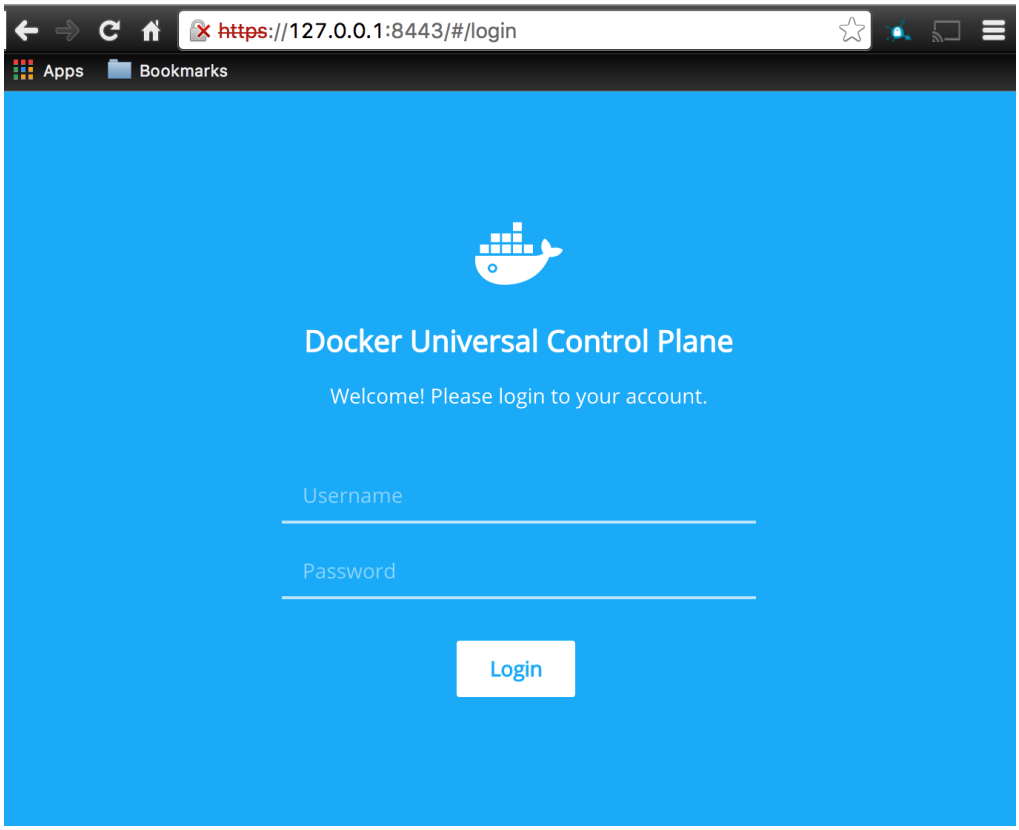
```
1  class ucpconfig::master(
2
3    $ucp_url         = $ucpconfig::ucp_url,
4    $ucp_username    = $ucpconfig::ucp_username,
5    $ucp_password    = $ucpconfig::ucp_password,
6    $ucp_fingerprint = $ucpconfig::ucp_fingerprint,
7
8  ) {
9
10 class { 'docker_ucp':
11   controller                => true,
12   host_address              => $::ipaddress_eth1,
13   version                   => '1.0.1',
14   usage                     => false,
15   tracking                  => false,
16   subject_alternative_names => $::ipaddress_eth0,
17   external_ca               => false,
18   swarm_scheduler           => 'binpack',
19   swarm_port                => 19001,
20   controller_port           => 443,
21   preserve_certs            => true,
22   docker_socket_path        => '/var/run/docker.sock',
23   license_file              => '/etc/docker/subscription.lic',
24   require                   => Class['docker']
25   }
26 }
```

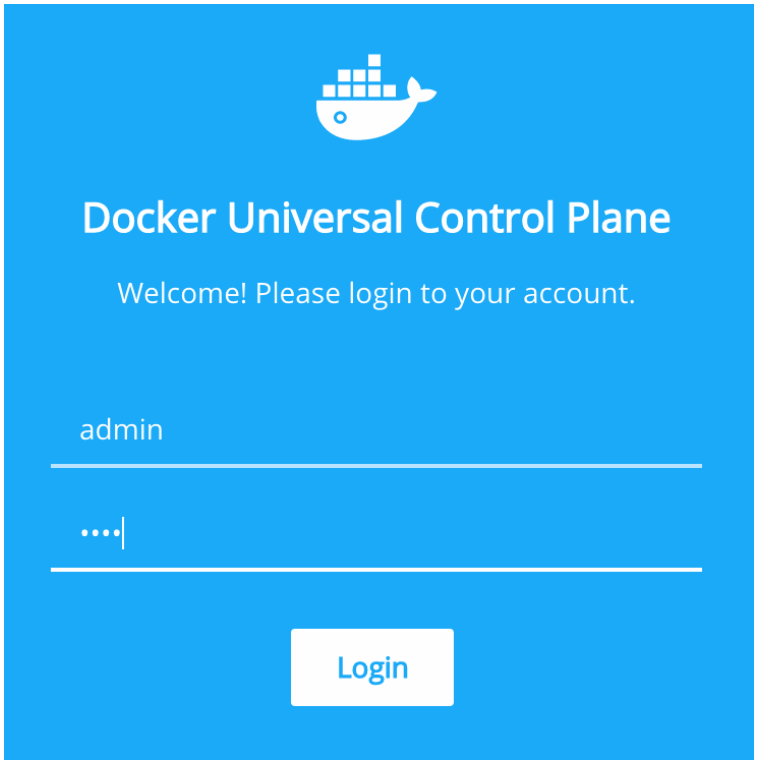
```
class ucpconfig::node (

  $ucp_url         = $ucpconfig::ucp_url,
  $ucp_username    = $ucpconfig::ucp_username,
  $ucp_password    = $ucpconfig::ucp_password,
  $ucp_fingerprint = $ucpconfig::ucp_fingerprint,

){

class { 'docker_ucp':
  ucp_url                  => $ucp_url,
  fingerprint              => $ucp_fingerprint,
  username                 => $ucp_username,
  password                 => $ucp_password ,
  host_address             => $::ipaddress_eth1,
  subject_alternative_names => $::ipaddress_eth0,
  replica                  => true,
  version                  => '1.0.1',
  usage                    => false,
  tracking                 => false,
  require                  => Class['docker']
  }
}

```

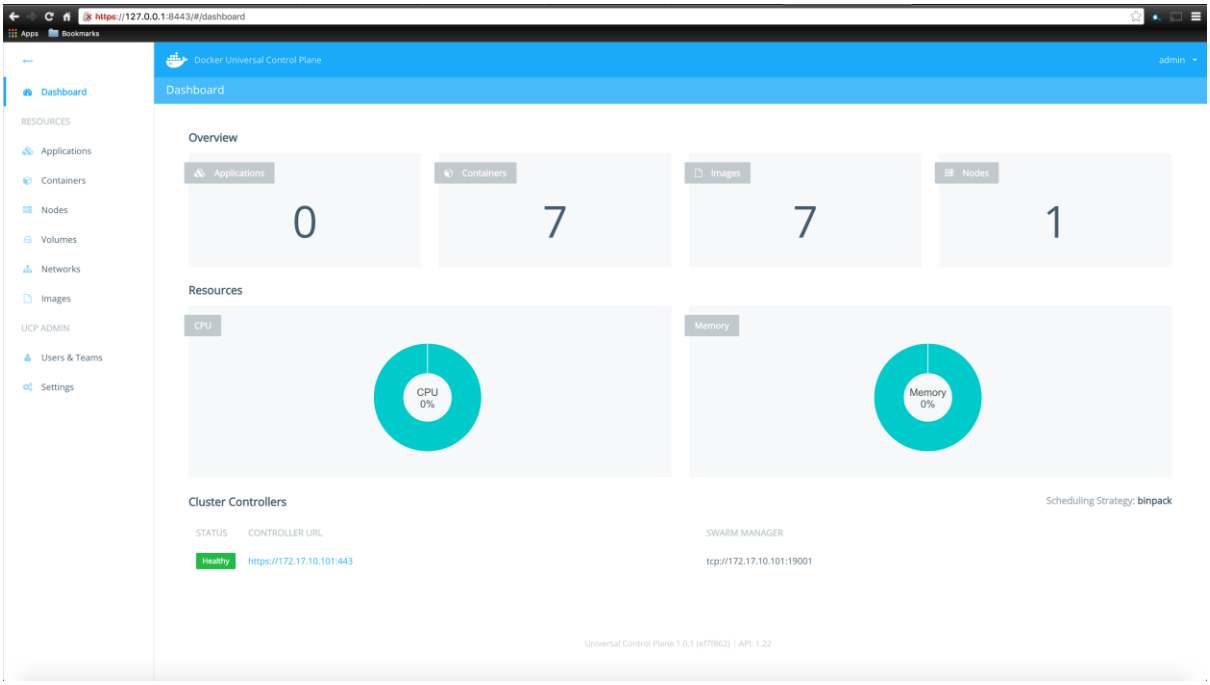
```
node 'ucp-01' {
  include ucpconfig
}

node 'ucp-02' {
  include ucpconfig
}

node 'ucp-03' {
  include ucpconfig

}
```

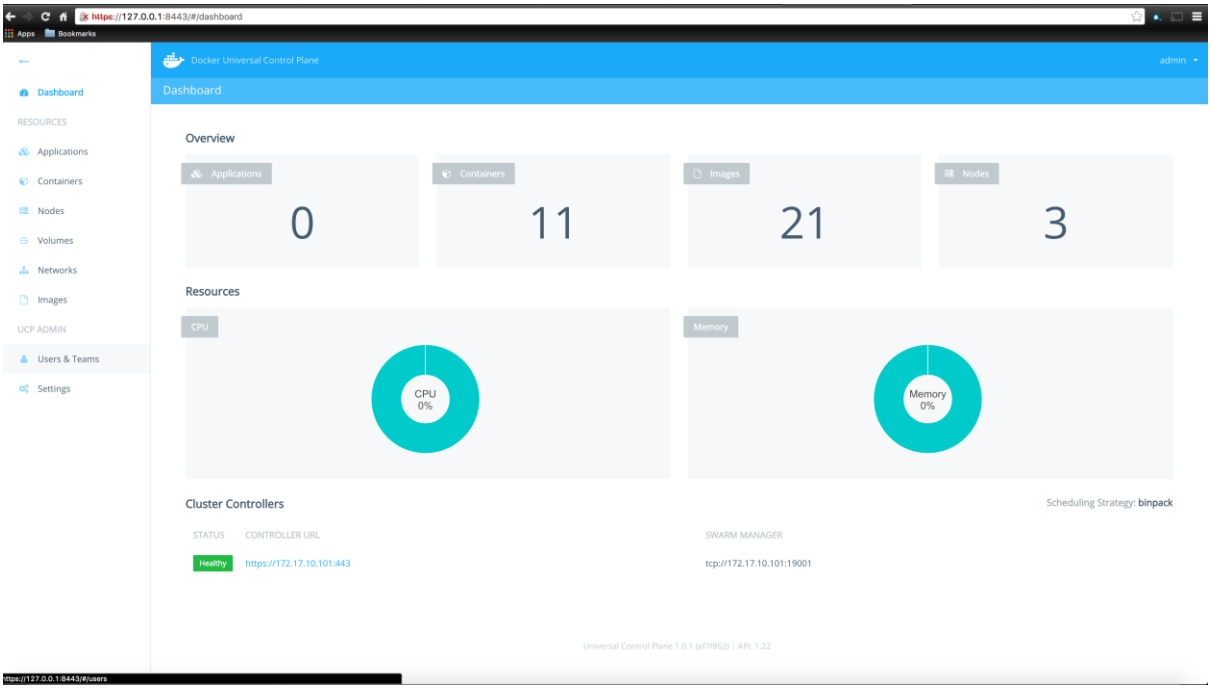


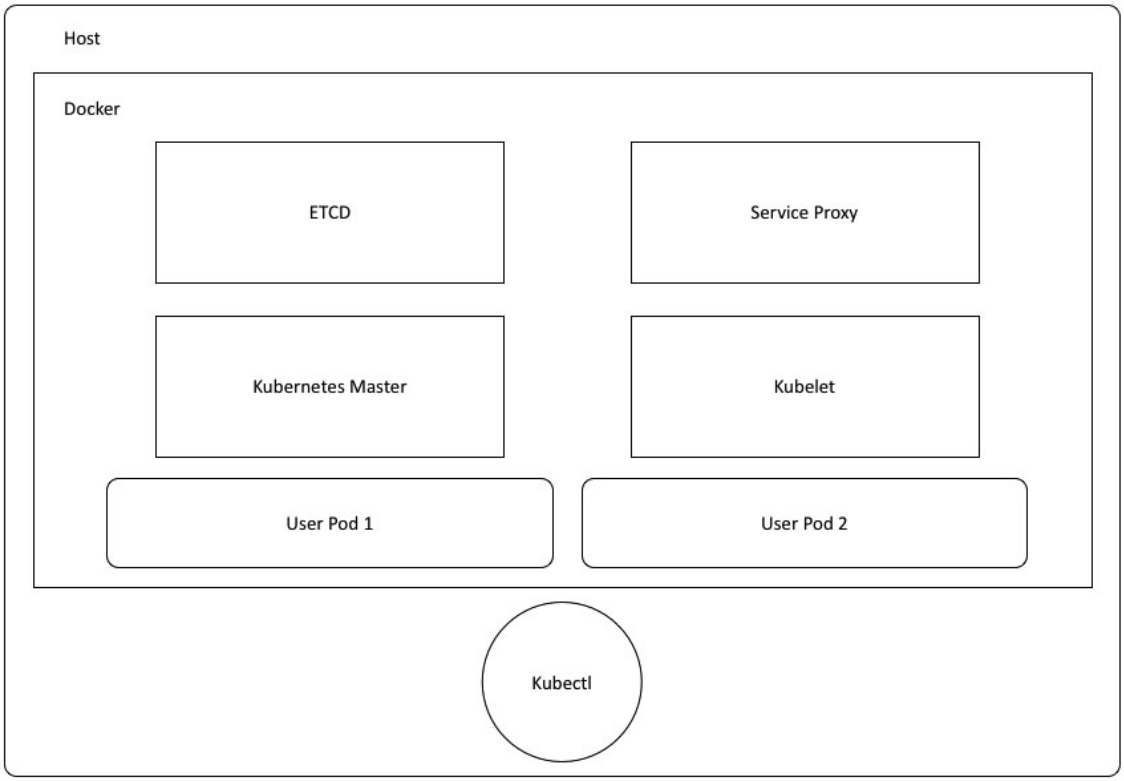

```
1  ucpconfig::ucp_url: https://172.17.10.101
2  ucpconfig::ucp_fingerprint: C2:7C:BB:C8:CF:26:59:0F:DB:BB:11:BC:02:18:C4:A4:18:C4:05:4E
3
```


https://127.0.0.1:8443/#/login
Apps
Bookmarks
Docker Universal Control Plane
Welcome! Please login to your account.
Username
Password
Login


Docker Universal Control Plane
Welcome! Please login to your account.
admin
Login


https://127.0.0.1:8443/#/dashboard
Apps
Bookmarks
Docker Universal Control Plane
admin
Dashboard
RESOURCES
Applications
Containers
Nodes
Volumes
Networks
Images
UCP ADMIN
Users & Teams
Settings
Overview
Applications
0
Containers
7
Images
7
Nodes
1
Resources
CPU
CPU
0%
Memory
Memory
0%
Cluster Controllers
Scheduling Strategy: binpack
STATUS
CONTROLLER URL
SWARM MANAGER
Healthy
https://172.17.10.101:443
tcp://172.17.10.101:19001
Universal Control Plane 1.0.1 (ef7f862) | API: 1.22


https://127.0.0.1:8443/#/dashboard
Apps
Bookmarks
Docker Universal Control Plane
admin
Dashboard
Dashboard
RESOURCES
Applications
Containers
Nodes
Volumes
Networks
Images
UCP ADMIN
Users & Teams
Settings
Overview
Applications
0
Containers
11
Images
21
Nodes
3
Resources
CPU
CPU
0%
Memory
Memory
0%
Cluster Controllers
Scheduling Strategy: binpack
STATUS
CONTROLLER URL
SWARM MANAGER
Healthy
https://172.17.10.101:443
tcp://172.17.10.101:19001
Universal Control Plane 1.0.1 (ef7f862) | API: 1.22
https://127.0.0.1:8443/#/users




```
---
-
  box: scottyc/centos-7-puppet-kernel-4-4
  cpu: 1
  ip: "172.17.9.101"
  name: kubernetes
  forward_ports:
      - { guest: 80, host: 8080 }
  ram: 2048
  shell_commands:
      - { shell: 'systemctl stop firewalld && systemctl disable firewalld'}
      - { shell: 'yum install -y wget git lvm2 device-mapper-libs' }
      - { shell: 'echo -e "172.17.9.101 kubernetes">/etc/hosts'}
      - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
      - { shell: 'cp /home/vagrant/kubernetes/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
      - { shell: 'cp /home/vagrant/kubernetes/modules/* -R /tmp/modules' }
```

```ruby
#!/usr/bin/ruby env

require "socket"
require 'resolv'

forge 'http://forge.puppetlabs.com'

mod 'puppetlabs/stdlib'
mod 'garethr/docker', :git => "https://github.com/scotty-c/garethr-docker.git"
mod 'maestrodev/wget'

```

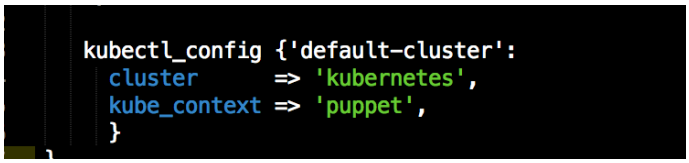
```
---
etcd:
    image: gcr.io/google_containers/etcd:2.2.1
    net: host
    command: ['/usr/local/bin/etcd', '--bind-addr=0.0.0.0:4001', '--data-dir=/var/etcd/data']
    restart: always
hyperkube:
    image: gcr.io/google_containers/hyperkube:v1.1.8
    volumes:
        - /:/rootfs:ro
        - /sys:/sys:ro
        - /dev:/dev
        - /var/lib/docker/:/var/lib/docker:ro
        - /var/lib/kubelet/:/var/lib/kubelet:rw
        - /var/run:/var/run:rw
        - /kubeconfig:/etc/kubernetes/manifests:ro
    net: host
    pid: host
    privileged: true
    restart: always
    command: ['/hyperkube', 'kubelet', '--containerized', '--address=0.0.0.0', "--api-servers=http://<%= @master_ip %>:8080", '--config=/etc/kubernetes/manifests']
proxy:
    image: gcr.io/google_containers/hyperkube:v1.1.8
    net: host
    pid: host
    privileged: true
    restart: always
    command: ['/hyperkube', 'proxy', "--master=http://<%= @master_ip %>:8080", '--v=2']
```

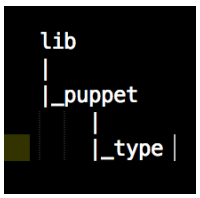
```
{
"apiVersion": "v1",
"kind": "Pod",
"metadata": {"name":"k8s-master"},
"spec":{
  "hostNetwork": true,
  "containers":[
    {
      "name": "controller-manager",
      "image": "gcr.io/google_containers/hyperkube:v1.1.8",
      "command": [
              "/hyperkube",
              "controller-manager",
              "--master=<%= @master_ip %>:8080",
              "--v=2"
        ]
    },
    {
      "name": "apiserver",
      "image": "gcr.io/google_containers/hyperkube:v1.1.8",
      "command": [
              "/hyperkube",
              "apiserver",
              "--portal-net=10.0.0.1/24",
              "--address=<%= @master_ip %>",
              "--etcd_servers=http://<%= @master_ip %>:4001",
              "--cluster_name=kubernetes",
              "--v=2"
        ]
    },
    {
      "name": "scheduler",
      "image": "gcr.io/google_containers/hyperkube:v1.1.8",
      "command": [
              "/hyperkube",
              "scheduler",
              "--master=<%= @master_ip %>:8080",
              "--v=2"
        ]
    }
  ]
 }
}
```

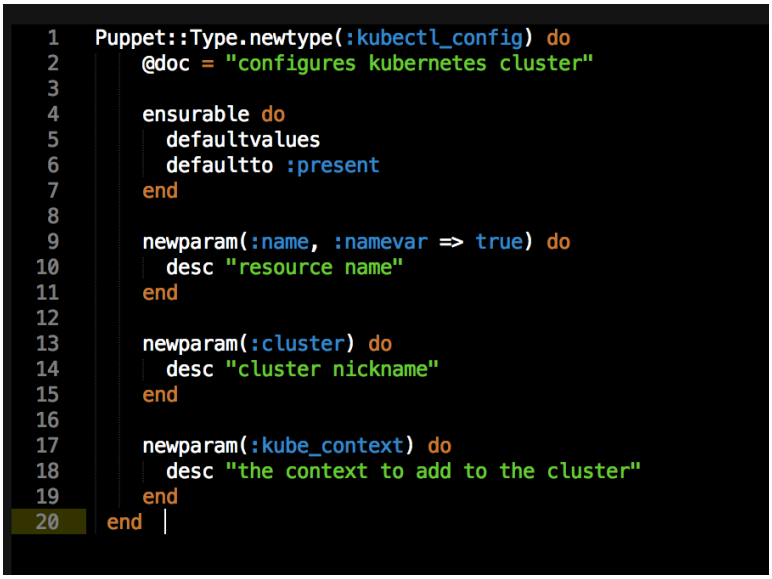
```
class kubernetes_docker (

  $master_ip = $kubernetes_docker::params::master_ip,

  ) inherits kubernetes_docker::params {

  include kubernetes_docker::install
  contain kubernetes_docker::config
  contain kubernetes_docker::apps
}
```

```
1  class kubernetes_docker::install {
2
3    package { 'device-mapper-libs':
4      ensure => installed,
5      }
6
7    class { 'docker':
8      tcp_bind    => 'tcp://127.0.0.1:4243',
9      socket_bind => 'unix:///var/run/docker.sock',
10     require => Package['device-mapper-libs']
11     } ->
12
13   file { '/kubeconfig':
14     ensure => directory,
15     group  => 'docker',
16     mode   => '0770',
17     } ->
18
19   file { '/kubeconfig/master.json':
20     ensure  => file,
21     content => template('kubernetes_docker/master.json.erb'),
22     mode    => '0755',
23     } ->
24
25   file { '/root/docker-compose.yml':
26     ensure  => file,
27     content => template('kubernetes_docker/docker-compose.yml.erb'),
28     } ->
29
30   docker_compose { kubernetes :
31     ensure => present,
32     source => '/root',
33     scale  => ['1', '1', '1']
34     }
35 }
```

```
class kubernetes_docker::config {

  wget::fetch { 'kubectl':
    source      => 'https://storage.googleapis.com/kubernetes-release/release/v1.1.8/bin/linux/amd64/kubectl',
    destination => '/usr/bin/kubectl',
    require     => Class['kubernetes_docker::install']
    } ->

  file { '/usr/bin/kubectl':
    mode   => '0777',
    } ->

   kubectl_config {'default-cluster':
     cluster      => 'kubernetes',
     kube_context => 'puppet',
     }
}
```

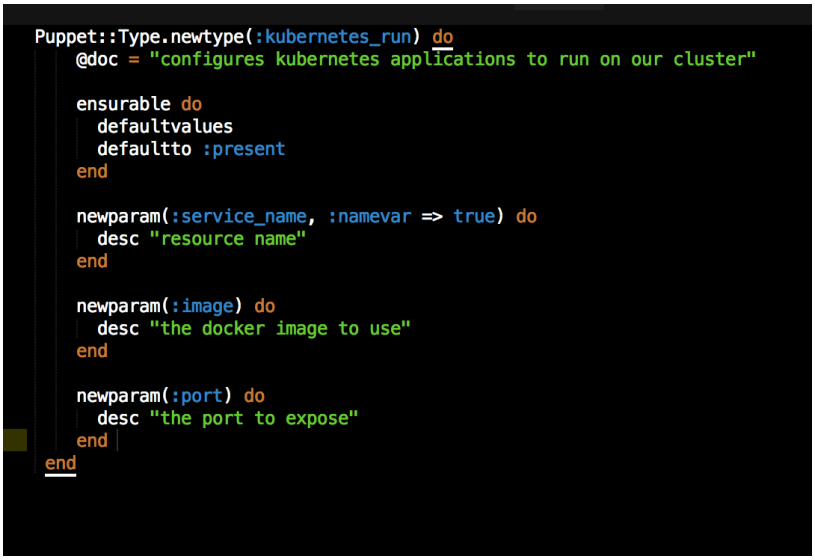





Puppet::Type.newtype(:kubectl_config) do
@doc = "configures kubernetes cluster"
ensurable do
defaultvalues
defaultto :present
end
newparam(:name, :namevar => true) do
desc "resource name"
end
newparam(:cluster) do
desc "cluster nickname"
end
newparam(:kube_context) do
desc "the context to add to the cluster"
end
end

```ruby
require 'socket'
require 'resolv'
require 'fileutils'

Puppet::Type.type(:kubectl_config).provide(:ruby) do
  desc "support for configuring a kubernetes cluster"

  mk_resource_methods

  commands :kubectl => "kubectl"

  def interface
    hostname = Socket.gethostname
    IPSocket.getaddress(hostname)
  end

  def kube_conf_server
    run = ['config','set-cluster', "#{(resource[:cluster])}", "--server=http://#{interface}:8080", '--insecure-skip-tls-verify=true']
  end

  def kube_conf_context
    run = ['config','set-context', "#{(resource[:kube_context])}", "--cluster=#{(resource[:cluster])}"]
  end

  def kube_conf_use
    run = ['config','use-context', "#{(resource[:kube_context])}"]
  end

  def exists?
    Puppet.info("checking if kubectl is configured")
    File.exist?('/.kube/config')
  end

  def create
    Puppet.info("configuring kubernetes cluster")
    kubectl *kube_conf_server
    kubectl *kube_conf_context
    kubectl *kube_conf_use
    FileUtils.ln_s('/.kube', '/root/.kube')
  end

  def destroy
    Puppet.info("removing kubectl config")
    FileUtils.rm_rf('/.kube')
  end
end
```

```
class kubernetes_docker::apps {

  kubernetes_run { 'nginx':
    image => 'nginx',
    port  => '80',
    require => Class['kubernetes_docker::config']
    }
}
```


Puppet::Type.newtype(:kubernetes_run) do
@doc = "configures kubernetes applications to run on our cluster"
ensurable do
defaultvalues
defaultto :present
end
newparam(:service_name, :namevar => true) do
desc "resource name"
end
newparam(:image) do
desc "the docker image to use"
end
newparam(:port) do
desc "the port to expose"
end
end

```ruby
Puppet::Type.type(:kubernetes_run).provide(:ruby) do
  desc "support for configuring a kubernetes cluster"

  mk_resource_methods

  commands :kubectl => "kubectl"
  commands :docker => "docker"

  def kube_run
    run = ['run', "#{(resource[:service_name])}", "--image=#{(resource[:image])}", "--port=#{(resource[:port])}"]
  end

  def kube_expose
    run = ['expose', 'rc', "#{(resource[:service_name])}", "--port=#{(resource[:port])}", "--external-ip=#{interface}"]
  end

  def exists?
    Puppet.info("checking kubectl if svc #{(resource[:service_name])} is configured")
    begin
      exists_args = ['get', 'svc']
      run = kubectl *exists_args
      run.match("#{(resource[:service_name])}")
    rescue => e
      return false
    end
  end

  def create
    Puppet.info("running #{(resource[:service_name])} on kubernetes cluster")
    begin
      args = ['get', 'nodes']
      kubectl *args
    rescue => e
      retry
    ensure
      kubectl *kube_run
      kubectl *kube_expose
    end
  end

  def destroy
    Puppet.info("removing application")
    destroy_args = ['rm', '-f', "#{(resource[:service_name])}"]
    docker *destroy_args
  end
end
```

```
1  node 'kubernetes' {
2
3    include kubernetes_docker
4
5  }
6
```

```
    kubernetes: Running: inline script
==> kubernetes: Running provisioner: puppet...
==> kubernetes: Running Puppet with environment production...
==> kubernetes: Info: Loading facts
==> kubernetes: Notice: Compiled catalog for localhost in environment production in 1.05 seconds
==> kubernetes: Info: Applying configuration version '1458034261'
==> kubernetes: Notice: /Stage[main]/Docker::Repos/Yumrepo[docker]/ensure: created
==> kubernetes: Info: changing mode of /etc/yum.repos.d/docker.repo from 600 to 644
==> kubernetes: Notice: /Stage[main]/Docker::Install/Package[docker]/ensure: created
==> kubernetes: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]/ensure: created
==> kubernetes: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage-setup]: Scheduling refresh of Service[docker]
==> kubernetes: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d]/ensure: created
==> kubernetes: Notice: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]/ensure: created
==> kubernetes: Info: /Stage[main]/Docker::Service/File[/etc/systemd/system/docker.service.d/service-overrides.conf]: Scheduling refresh of Exec[docker-systemd-reload-before-service]
==> kubernetes: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]/returns: executed successfully
==> kubernetes: Notice: /Stage[main]/Docker::Service/Exec[docker-systemd-reload-before-service]: Triggered 'refresh' from 1 events
==> kubernetes: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]/ensure: created
==> kubernetes: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker-storage]: Scheduling refresh of Service[docker]
==> kubernetes: Notice: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]/ensure: created
==> kubernetes: Info: /Stage[main]/Docker::Service/File[/etc/sysconfig/docker]: Scheduling refresh of Service[docker]
==> kubernetes: Notice: /Stage[main]/Docker::Service/Service[docker]/ensure: ensure changed 'stopped' to 'running'
==> kubernetes: Info: /Stage[main]/Docker::Service/Service[docker]: Unscheduling refresh on Service[docker]
==> kubernetes: Notice: /Stage[main]/Docker::Compose/Exec[Install Docker Compose 1.6.0]/returns: executed successfully
==> kubernetes: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose-1.6.0]/mode: mode changed '0644' to '0755'
==> kubernetes: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose]/ensure: created
==> kubernetes: Notice: /Stage[main]/Kubernetes_docker::Install/File[/kubeconfig]/ensure: created
==> kubernetes: Notice: /Stage[main]/Kubernetes_docker::Install/File[/kubeconfig/master.json]/ensure: defined content as '{md5}f3fd255926e4113cef30af4e65906e5e'
==> kubernetes: Notice: /Stage[main]/Kubernetes_docker::Install/File[/root/docker-compose.yml]/ensure: defined content as '{md5}08cdcd9d97c6db3cbeca1d3d87175da6'
==> kubernetes: Info: Checking if docker-compose.yml exists
==> kubernetes: Info: bring up containers
==> kubernetes: Notice: /Stage[main]/Kubernetes_docker::Install/Docker_compose[kubernetes]/ensure: created
==> kubernetes: Notice: /Stage[main]/Kubernetes_docker::Config/Wget::Fetch[kubectl]/Exec[wget-kubectl]/returns: executed successfully
==> kubernetes: Notice: /Stage[main]/Kubernetes_docker::Config/File[/usr/bin/kubectl]/mode: mode changed '0644' to '0777'
==> kubernetes: Info: checking if kubectl is configured
==> kubernetes: Info: configuring kubernetes cluster
==> kubernetes: Notice: /Stage[main]/Kubernetes_docker::Config/Kubectl_config[default-cluster]/ensure: created
==> kubernetes: Info: checking kubectl if svc nginx is configured
==> kubernetes: Info: running nginx on kubernetes cluster
==> kubernetes: Notice: /Stage[main]/Kubernetes_docker::Apps/Kubernetes_run[nginx]/ensure: created
==> kubernetes: Notice: Finished catalog run in 165.10 seconds
```

```
[root@kubernetes ~]# kubectl get svc
NAME         CLUSTER_IP   EXTERNAL_IP    PORT(S)   SELECTOR    AGE
kubernetes   10.0.0.1     <none>         443/TCP   <none>      1h
nginx        10.0.0.50    172.17.9.101   80/TCP    run=nginx   1h
[root@kubernetes ~]#
```

Chapter 8: Logging, Monitoring, and Recovery Techniques

```
#!/usr/bin/ruby env

require "socket"
$hostname = Socket.gethostname

forge 'http://forge.puppetlabs.com'


mod 'puppetlabs/stdlib'
mod 'puppetlabs/vcsrepo'
mod 'nanliu/staging'
mod 'KyleAnderson/consul'
mod 'scottyc/docker_swarm'
mod 'scottyc/golang'
mod 'garethr/docker', :git => "https://github.com/scotty-c/garethr-docker.git"
mod 'stankevich/python'
mod 'stahnma/epel'
mod 'maestrodev/wget'


```

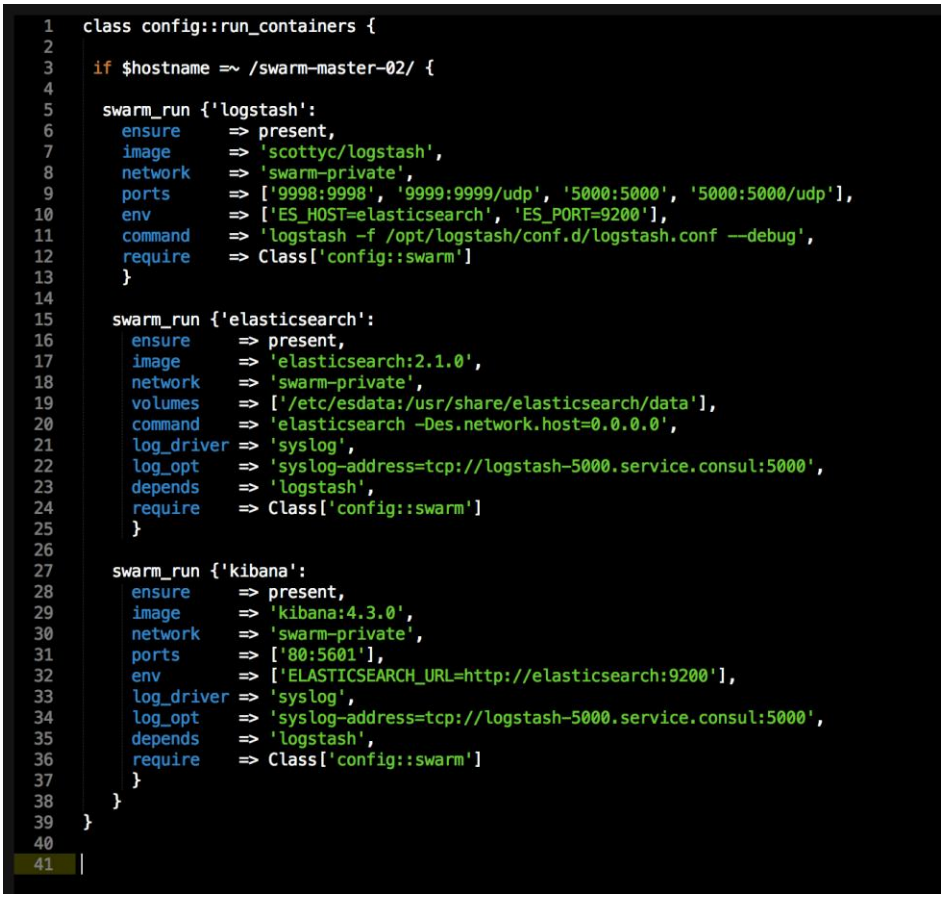
```
docker_swarm::swarm_version: v1.1.3
docker_swarm::backend: consul
docker_swarm::backend_ip: 172.17.8.101
docker_swarm::backend_port: 8500
docker_swarm::advertise_int: enp0s8
consul::version: 0.6.3
```

```
class config::compose {

if $hostname =~ /^swarm-master*/ {

 notice ["This server is the Swarm Manager."]

}

else {

file { '/root/docker-compose.yml':
  ensure  => file,
  content => template("config/registrator.yml.erb"),
  } ->

docker_compose {'swarm app':
  ensure => present,
  source => '/root',
  scale => ['1']
    }
  }
}
```

```
class config::consul_config {

if $hostname =~ /^*101*$/ {

  class { 'consul':
    config_hash => {
      'datacenter'       => 'dev',
      'data_dir'         => '/opt/consul',
      'ui_dir'           => '/opt/consul/ui',
      'bind_addr'        => $::ipaddress_enp0s8,
      'client_addr'      => '0.0.0.0',
      'node_name'        => "$::hostname",
      'advertise_addr'   => '172.17.8.101',
      'bootstrap_expect' => '1',
      'server'           => true
      }
    }
  }

else {

 class { 'consul':
  config_hash => {
      'bootstrap'      => false,
      'datacenter'     => 'dev',
      'data_dir'       => '/opt/consul',
      'ui_dir'         => '/opt/consul/ui',
      'bind_addr'      => $::ipaddress_enp0s8,
      'client_addr'    => '0.0.0.0',
      'node_name'      => "$::hostname",
      'advertise_addr' => $::ipaddress_enp0s8,
      'start_join'     => ['172.17.8.101','172.17.8.103','172.17.8.103'],
      'server'         => false
      }
    }
  }

  consul::service { 'docker-service':
  checks  => [
    {
    script   => 'service docker status',
    interval => '10s',
    tags     => ['docker-service']
    }
  ],
  address => $::ipaddress_enp0s8,
  }


 }
```

```
class config::dns {

  package { 'bind':
    ensure => present
  } ->

  file { '/etc/named.conf':
    ensure  => present,
    content => template("config/named.conf.erb"),
    mode    => '0644',
    owner   => 'root',
    group   => 'root',
    require => Package['bind'],
  } ~>

  file { '/etc/named/consul.conf':
    ensure  => present,
    content => template("config/consul.conf.erb"),
    mode    => '0644',
    owner   => 'root',
    group   => 'root',
    require => Package['bind'],
  } ~>

  service { 'named':
    ensure  => running,
    enable  => true,
    require => File['/etc/named.conf'],
  }
}
```

```
# == Class: config
#
#
class config(

  $consul_ip = "$::ipaddress_enp0s8",

){

  include config::consul_config
  contain config::swarm
  contain config::compose
  contain config::dns
  contain config::run_containers

  Class['config::swarm'] -> Class['config::compose'] -> Class['config::dns'] -> Class['config::run_containers']
}
```



```
class config::run_containers {

 if $hostname =~ /swarm-master-02/ {

  swarm_run {'logstash':
    ensure     => present,
    image      => 'scottyc/logstash',
    network    => 'swarm-private',
    ports      => ['9998:9998', '9999:9999/udp', '5000:5000', '5000:5000/udp'],
    env        => ['ES_HOST=elasticsearch', 'ES_PORT=9200'],
    command    => 'logstash -f /opt/logstash/conf.d/logstash.conf --debug',
    require    => Class['config::swarm']
    }

   swarm_run {'elasticsearch':
     ensure     => present,
     image      => 'elasticsearch:2.1.0',
     network    => 'swarm-private',
     volumes    => ['/etc/esdata:/usr/share/elasticsearch/data'],
     command    => 'elasticsearch -Des.network.host=0.0.0.0',
     log_driver => 'syslog',
     log_opt    => 'syslog-address=tcp://logstash-5000.service.consul:5000',
     depends    => 'logstash',
     require    => Class['config::swarm']
     }

   swarm_run {'kibana':
     ensure     => present,
     image      => 'kibana:4.3.0',
     network    => 'swarm-private',
     ports      => ['80:5601'],
     env        => ['ELASTICSEARCH_URL=http://elasticsearch:9200'],
     log_driver => 'syslog',
     log_opt    => 'syslog-address=tcp://logstash-5000.service.consul:5000',
     depends    => 'logstash',
     require    => Class['config::swarm']
     }
   }
}

```

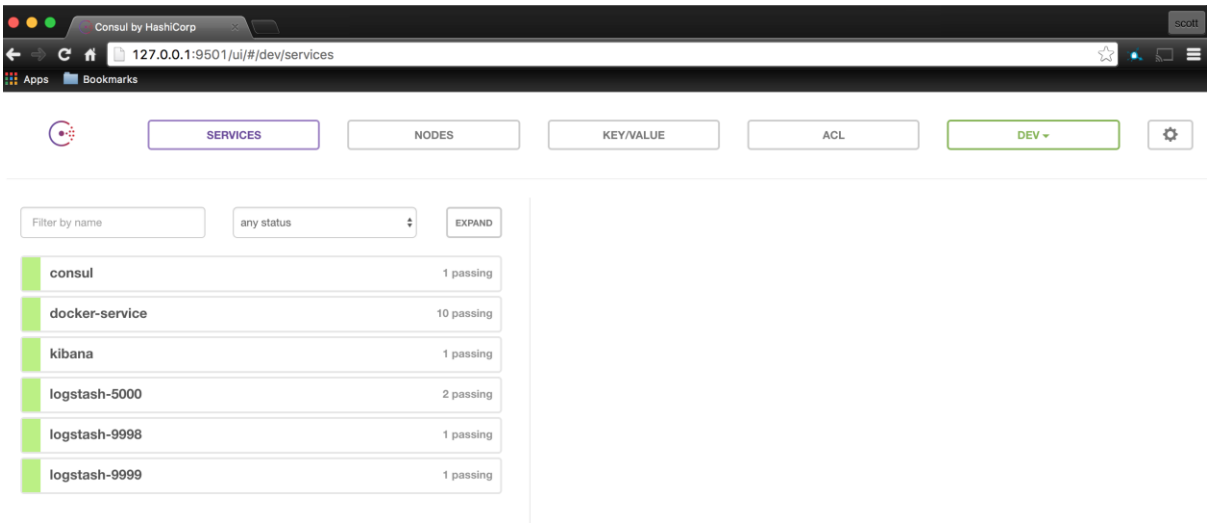
```
class config::swarm {

   class { 'docker_swarm':
   require => Class['config::consul_config']
   }

   docker_network { 'swarm-private':
   ensure => present,
   create => true,
   driver => 'overlay',
   require => Class['config::consul_config']
   }

   if $hostname =~ /^swarm-master*/ {

   swarm_cluster {'cluster 1':
     ensure       => present,
     backend      => 'consul',
     cluster_type => 'manage',
     port         => '8500',
     address      => '172.17.8.101',
     advertise    => $::ipaddress_enp0s8,
     path         => 'swarm',
     }
   }

   else {

   swarm_cluster {'cluster 1':
    ensure       => present,
    backend      => 'consul',
    cluster_type => 'join',
    port         => '8500',
    address      => '172.17.8.101',
    path         => 'swarm'
     }
   }
}
```

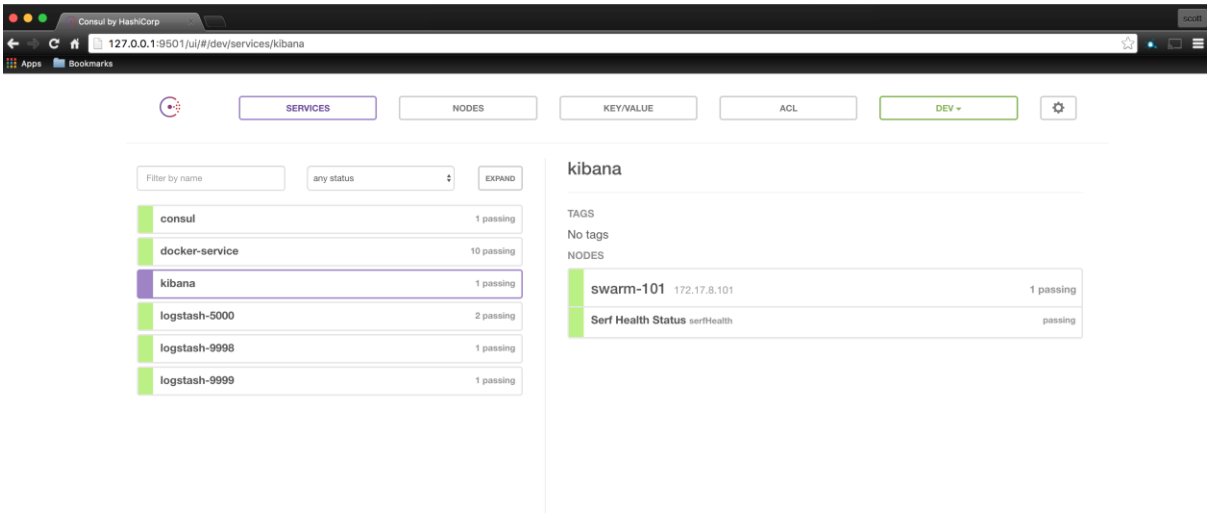
```
zone "consul" IN {
  type forward;
  forward only;
  forwarders { 127.0.0.1 port 8600; };
};
```

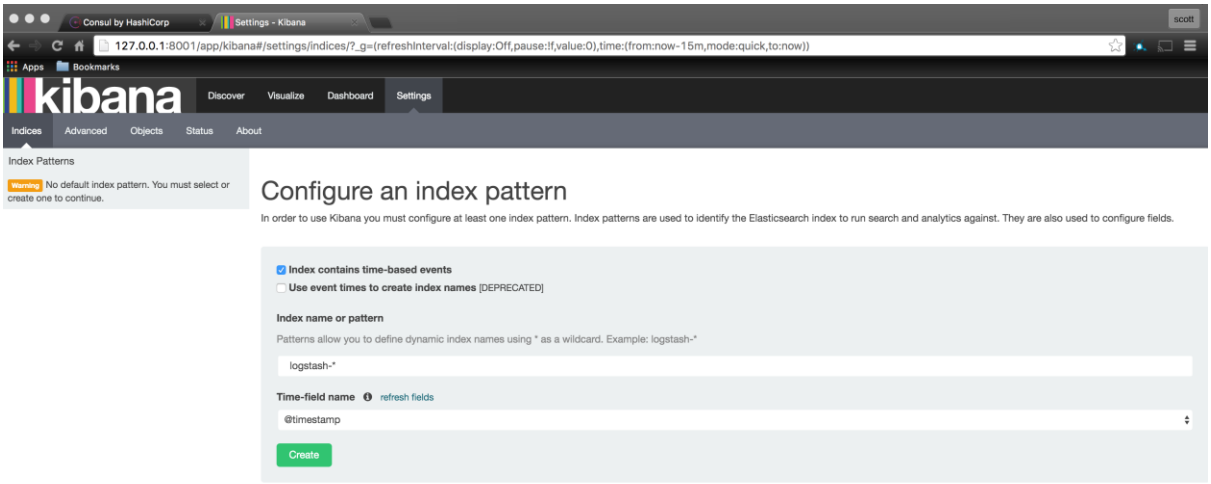
```
options {
  listen-on port 53 { 127.0.0.1; };
  listen-on-v6 port 53 { ::1; };
  directory       "/var/named";
  dump-file       "/var/named/data/cache_dump.db";
  statistics-file "/var/named/data/named_stats.txt";
  memstatistics-file "/var/named/data/named_mem_stats.txt";
  allow-query     { localhost; };
  recursion yes;

  dnssec-enable no;
  dnssec-validation no;

  /* Path to ISC DLV key */
  bindkeys-file "/etc/named.iscdlv.key";

  managed-keys-directory "/var/named/dynamic";
};

include "/etc/named/consul.conf";
```

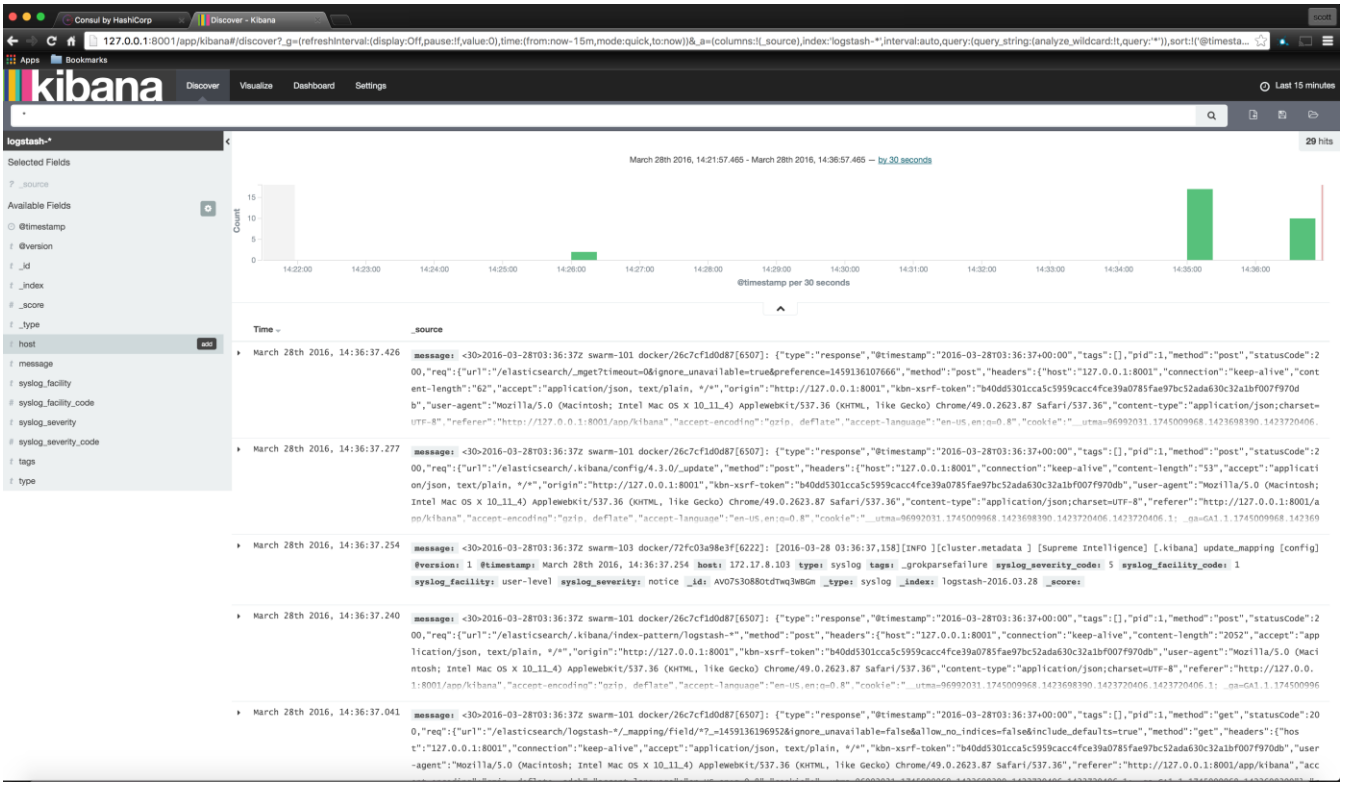
```
registrator:
  image: gliderlabs/registrator
  net: "host"
  volumes:
   - "/var/run/docker.sock:/tmp/docker.sock"
  command: consul://<%= @consul_ip %>:8500

```

```
node 'swarm-101' { include config }

node 'swarm-102' { include config }

node 'swarm-103' { include config }

node 'swarm-master-01' { include config }

node 'swarm-master-02' { include config }

```




Consul by HashiCorp
127.0.0.1:9501/ui/#/dev/services/kibana
Apps
Bookmarks
SERVICES
NODES
KEY/VALUE
ACL
DEV
Filter by name
any status
EXPAND
consul 1 passing
docker-service 10 passing
kibana 1 passing
logstash-5000 2 passing
logstash-9998 1 passing
logstash-9999 1 passing
kibana
TAGS
No tags
NODES
swarm-101 172.17.8.101 1 passing
Serf Health Status serfHealth passing


Consul by HashiCorp
Settings - Kibana
127.0.0.1:8001/app/kibana#/settings/indices/?_g=(refreshInterval:(display:Off,pause:!f,value:0),time:(from:now-15m,mode:quick,to:now))
Apps
Bookmarks
kibana
Discover
Visualize
Dashboard
Settings
Indices
Advanced
Objects
Status
About
Index Patterns
Warning No default index pattern. You must select or create one to continue.
Configure an index pattern
In order to use Kibana you must configure at least one index pattern. Index patterns are used to identify the Elasticsearch index to run search and analytics against. They are also used to configure fields.
Index contains time-based events
Use event times to create index names [DEPRECATED]
Index name or pattern
Patterns allow you to define dynamic index names using * as a wildcard. Example: logstash-*
logstash-*
Time-field name
refresh fields
@timestamp
Create




```
consul::service { 'docker-service':
checks  => [
  {
  script   => 'service docker status',
  interval => '10s',
  tags     => ['docker-service']
  }
],
address => $::ipaddress_enp0s8,
}
```

Service 'docker-service' check service:docker-service passing

NOTES

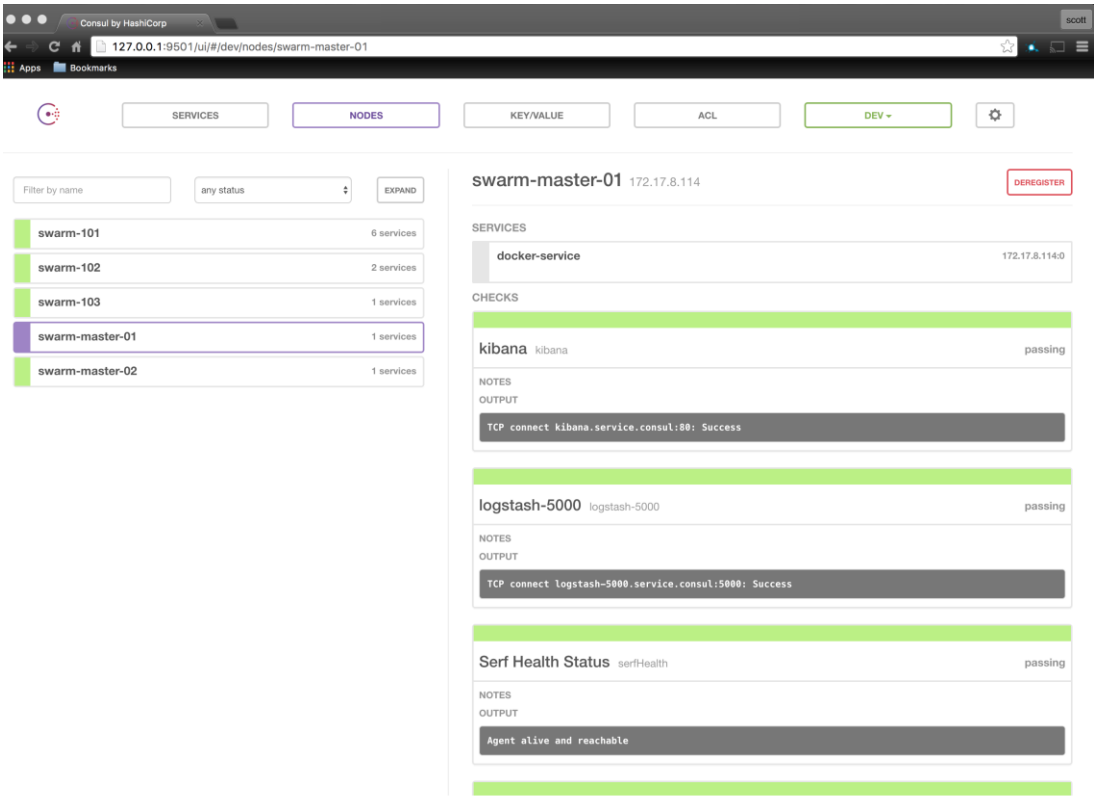
OUTPUT

```
Redirecting to /bin/systemctl status  docker.service
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; vendor preset: disabled)
  Drop-In: /etc/systemd/system/docker.service.d
           └─service-overrides.conf
   Active: active (running) since Mon 2016-03-28 07:33:57 UTC; 8min ago
     Docs: https://docs.docker.com
 Main PID: 6221 (docker)
   CGroup: /system.slice/docker.service
           └─6221 /usr/bin/docker daemon -H tcp://0.0.0.0:2375 -H unix:///var/run/docker.sock --cluste
```

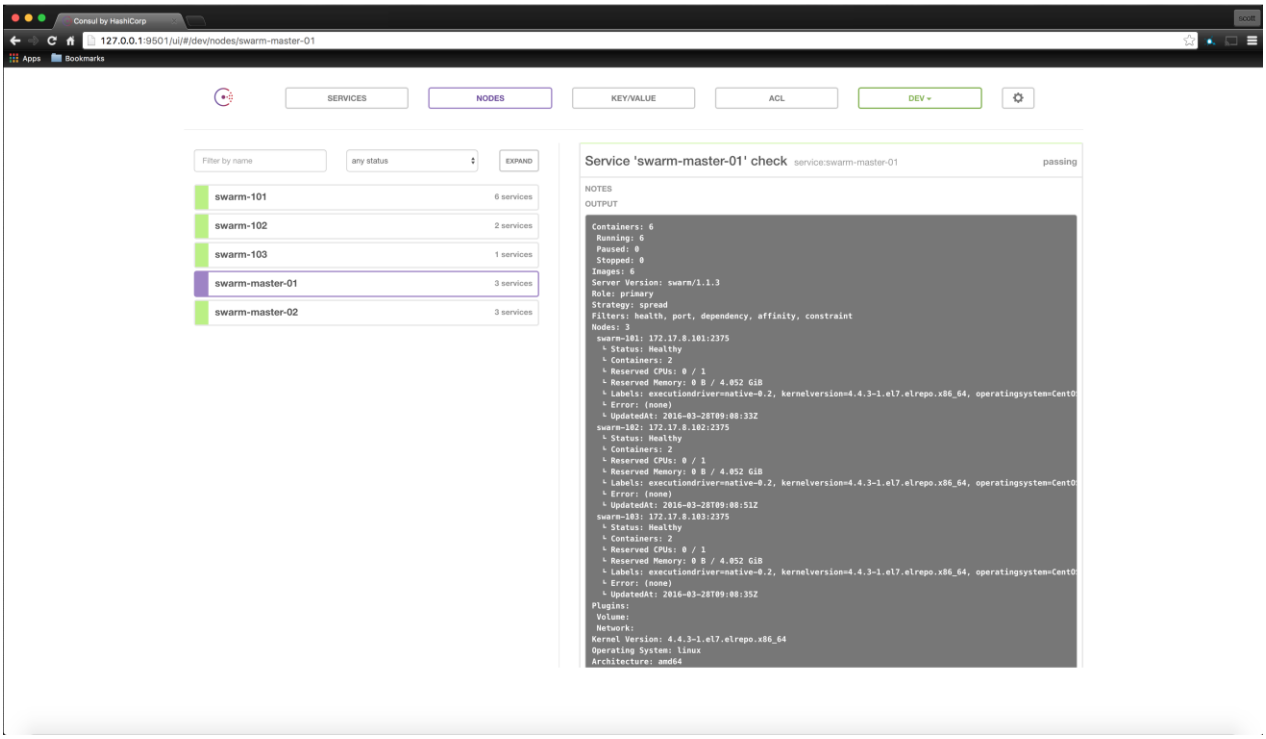
LOCK SESSIONS

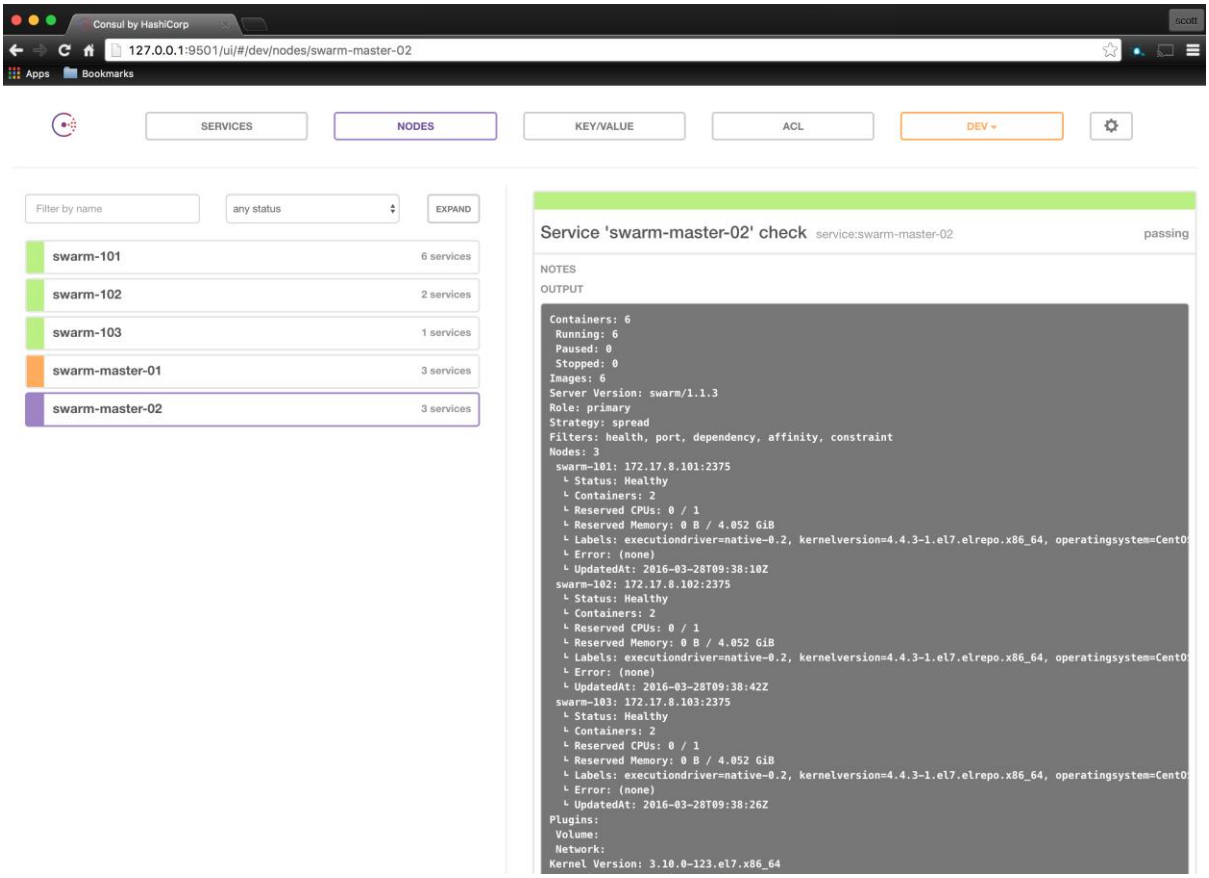
No sessions

```
if $hostname =~ /^swarm-master*/ {
  consul::check { 'kibana':
    ensure   =>  present,
    tcp      => 'kibana.service.consul:80',
    interval => '10s',
    }

  consul::check { 'logstash-5000':
    ensure   =>  present,
    tcp      => 'logstash-5000.service.consul:5000',
    interval => '10s',
    }
```

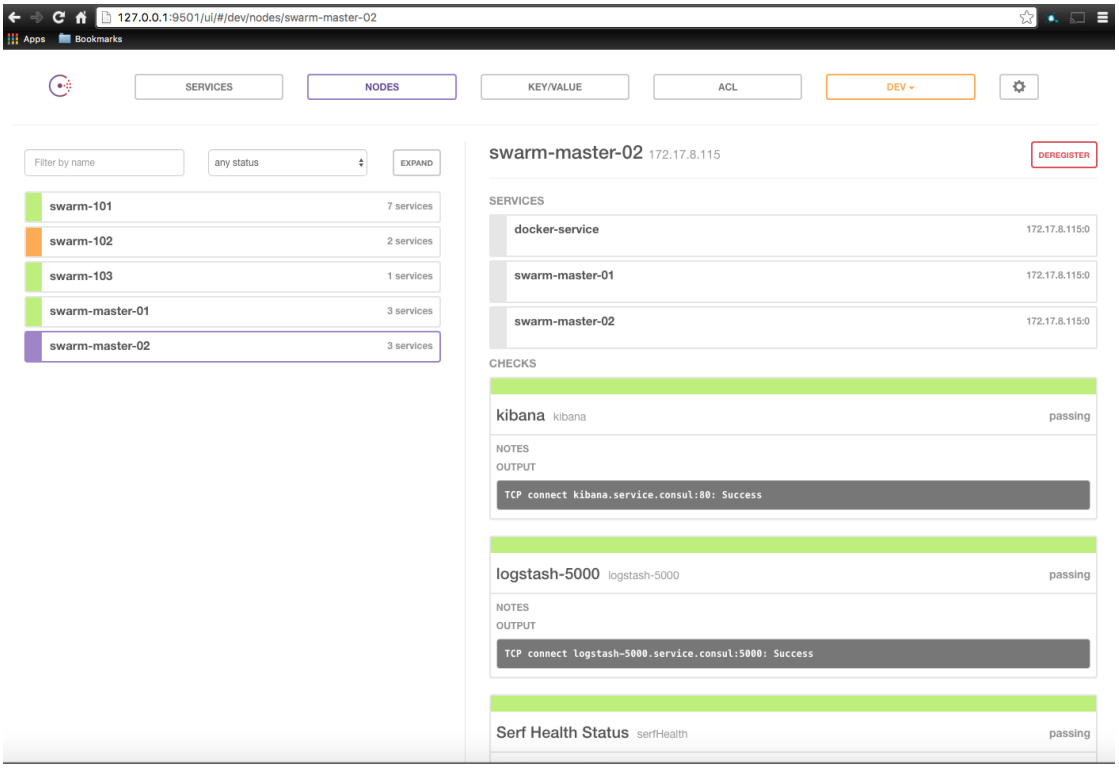


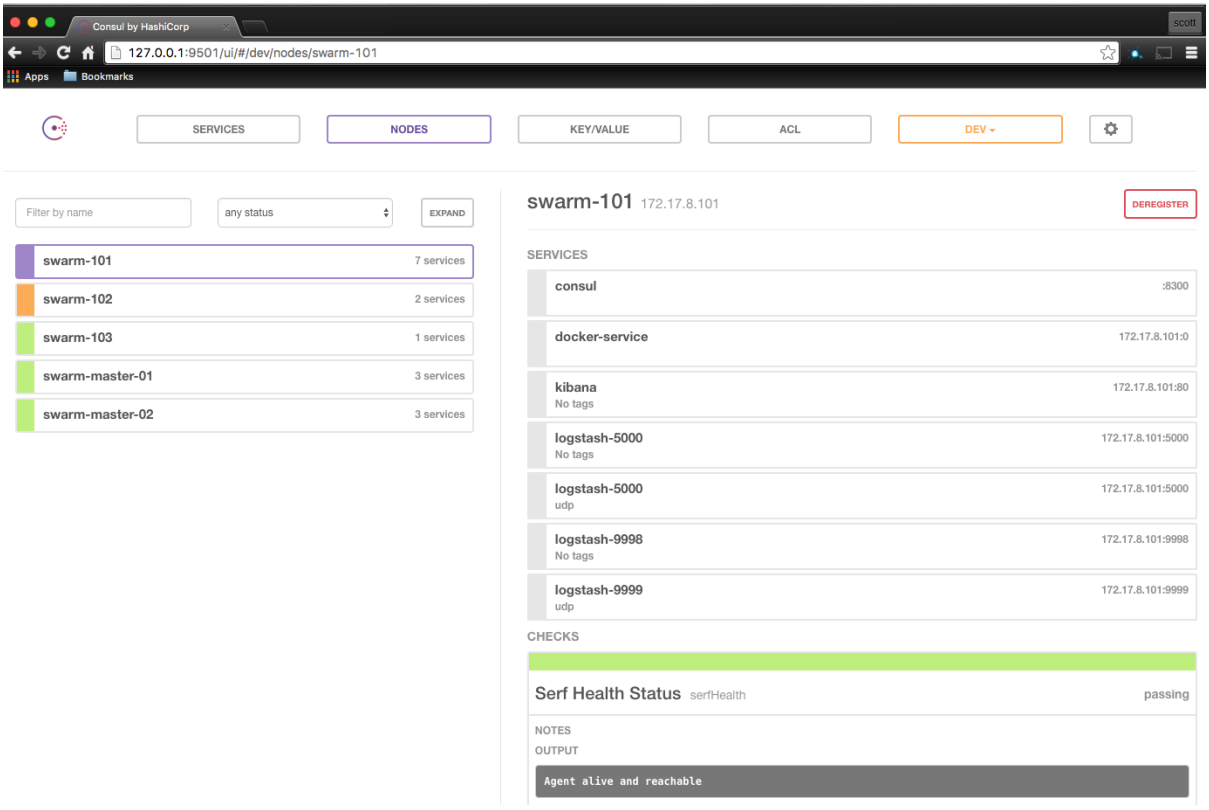

```
consul::service { 'swarm-master-01':
  checks  => [
    {
    script   => 'docker -H tcp://172.17.8.114:4000 info',
    interval => '10s',
    tags     => ['swarm-master-01']
    }
  ],
  address => $::ipaddress_enp0s8,
   }

consul::service { 'swarm-master-02':
  checks  => [
    {
    script   => 'docker -H tcp://172.17.8.115:4000 info',
    interval => '10s',
    tags     => ['swarm-master-02']
    }
  ],
  address => $::ipaddress_enp0s8,
   }
 }
}
```

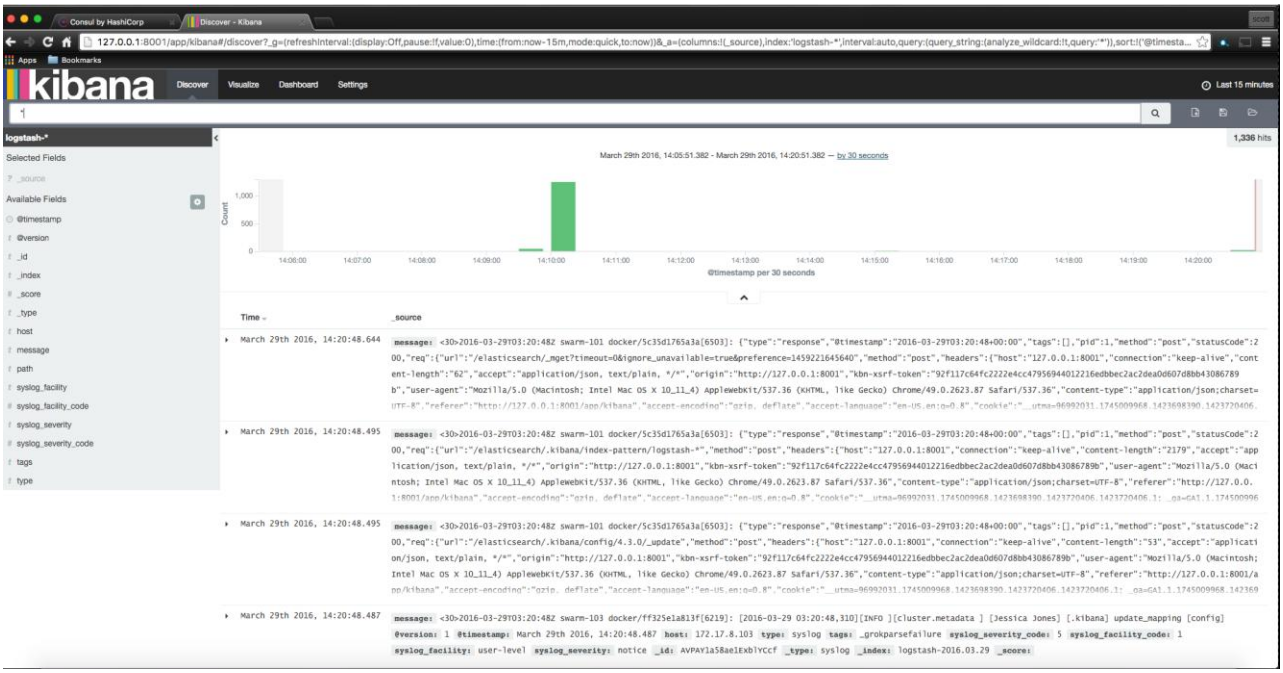

Consul by HashiCorp
127.0.0.1:9501/ui/#/dev/nodes/swarm-master-01
Apps
Bookmarks
SERVICES
NODES
KEY/VALUE
ACL
DEV
Filter by name
any status
EXPAND
swarm-101
6 services
swarm-102
2 services
swarm-103
1 services
swarm-master-01
3 services
swarm-master-02
3 services
Service 'swarm-master-01' check
service:swarm-master-01
passing
NOTES
OUTPUT
Containers: 6
Running: 6
Paused: 0
Stopped: 0
Images: 6
Server Version: swarm/1.1.3
Role: primary
Strategy: spread
Filters: health, port, dependency, affinity, constraint
Nodes: 3
swarm-101: 172.17.8.101:2375
└ Status: Healthy
└ Containers: 2
└ Reserved CPUs: 0 / 1
└ Reserved Memory: 0 B / 4.052 GiB
└ Labels: executiondriver=native-0.2, kernelversion=4.4.3-1.el7.elrepo.x86_64, operatingsystem=Cent
└ Error: (none)
└ UpdatedAt: 2016-03-28T09:08:33Z
swarm-102: 172.17.8.102:2375
└ Status: Healthy
└ Containers: 2
└ Reserved CPUs: 0 / 1
└ Reserved Memory: 0 B / 4.052 GiB
└ Labels: executiondriver=native-0.2, kernelversion=4.4.3-1.el7.elrepo.x86_64, operatingsystem=Cent
└ Error: (none)
└ UpdatedAt: 2016-03-28T09:08:51Z
swarm-103: 172.17.8.103:2375
└ Status: Healthy
└ Containers: 2
└ Reserved CPUs: 0 / 1
└ Reserved Memory: 0 B / 4.052 GiB
└ Labels: executiondriver=native-0.2, kernelversion=4.4.3-1.el7.elrepo.x86_64, operatingsystem=Cent
└ Error: (none)
└ UpdatedAt: 2016-03-28T09:08:35Z
Plugins:
Volume:
Network:
Kernel Version: 4.4.3-1.el7.elrepo.x86_64
Operating System: linux
Architecture: amd64



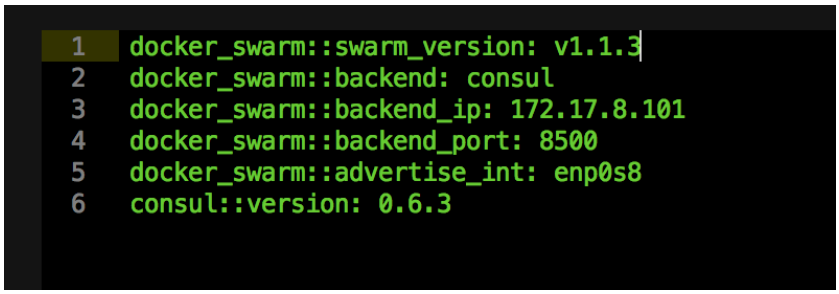

```
swarm_run {'kibana':
  ensure     => present,
  image      => 'kibana:4.3.0',
  network    => 'swarm-private',
  ports      => ['80:5601'],
  env        => ['ELASTICSEARCH_URL=http://elasticsearch:9200', 'reschedule:on-node-failure'],
  log_driver => 'syslog',
  log_opt    => 'syslog-address=tcp://logstash-5000.service.consul:5000',
  depends    => 'logstash',
  require    => Class['config::swarm']
  }
```


127.0.0.1:9501/ui/#/dev/nodes/swarm-master-02
Apps
Bookmarks
SERVICES
NODES
KEY/VALUE
ACL
DEV
Filter by name
any status
EXPAND
swarm-101 7 services
swarm-102 2 services
swarm-103 1 services
swarm-master-01 3 services
swarm-master-02 3 services
swarm-master-02 172.17.8.115
DEREGISTER
SERVICES
docker-service 172.17.8.115:0
swarm-master-01 172.17.8.115:0
swarm-master-02 172.17.8.115:0
CHECKS
kibana kibana passing
NOTES
OUTPUT
TCP connect kibana.service.consul:80: Success
logstash-5000 logstash-5000 passing
NOTES
OUTPUT
TCP connect logstash-5000.service.consul:5000: Success
Serf Health Status serfHealth passing


Consul by HashiCorp
scott
127.0.0.1:9501/ui/#/dev/nodes/swarm-101
Apps
Bookmarks
SERVICES
NODES
KEY/VALUE
ACL
DEV
Filter by name
any status
EXPAND
swarm-101 7 services
swarm-102 2 services
swarm-103 1 services
swarm-master-01 3 services
swarm-master-02 3 services
swarm-101 172.17.8.101
DEREGISTER
SERVICES
consul :8300
docker-service 172.17.8.101:0
kibana 172.17.8.101:80
No tags
logstash-5000 172.17.8.101:5000
No tags
logstash-5000 172.17.8.101:5000
udp
logstash-9998 172.17.8.101:9998
No tags
logstash-9999 172.17.8.101:9999
udp
CHECKS
Serf Health Status serfHealth passing
NOTES
OUTPUT
Agent alive and reachable


Consul by HashiCorp
Discover - Kibana
127.0.0.1:8001/app/kibana#/discover?_g=(refreshInterval:(display:Off,pause:!f,value:0),time:(from:now-15m,mode:quick,to:now))&_a=(columns:!(_source),index:'logstash-*',interval:auto,query:(query_string:(analyze_wildcard:!t,query:'*')),sort:!('@timesta...
Apps
Bookmarks
kibana
Discover
Visualize
Dashboard
Settings
Last 15 minutes
logstash-*
Selected Fields
_source
Available Fields
@timestamp
@version
_id
_index
_score
_type
host
message
path
syslog_facility
syslog_facility_code
syslog_severity
syslog_severity_code
tags
type
1,336 hits
March 29th 2016, 14:05:51.382 - March 29th 2016, 14:20:51.382 — by 30 seconds
Count
1,000
500
0
14:06:00
14:07:00
14:08:00
14:09:00
14:10:00
14:11:00
14:12:00
14:13:00
14:14:00
14:15:00
14:16:00
14:17:00
14:18:00
14:19:00
14:20:00
@timestamp per 30 seconds
Time
_source
March 29th 2016, 14:20:48.644
March 29th 2016, 14:20:48.495
March 29th 2016, 14:20:48.495
March 29th 2016, 14:20:48.487
message: <30>2016-03-29T03:20:48Z swarm-101 docker/5c35d1765a3a[6503]: {"type":"response","@timestamp":"2016-03-29T03:20:48+00:00","tags":[],"pid":1,"method":"post","statusCode":200,"req":{"url":"/elasticsearch/_mget?timeout=0&ignore_unavailable=true&preference=1459221645640","method":"post","headers":{"host":"127.0.0.1:8001","connection":"keep-alive","content-length":"62","accept":"application/json, text/plain, */*","origin":"http://127.0.0.1:8001","kbn-xsrf-token":"92f117c64fc2222e4cc47956944012216edbbec2ac2dea0d607d8bb43086789b","user-agent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/49.0.2623.87 Safari/537.36","content-type":"application/json;charset=UTF-8","referer":"http://127.0.0.1:8001/app/kibana","accept-encoding":"gzip, deflate","accept-language":"en-US,en;q=0.8","cookie":"__utma=96992031.1745009968.1423698390.1423720406.
message: <30>2016-03-29T03:20:48Z swarm-101 docker/5c35d1765a3a[6503]: {"type":"response","@timestamp":"2016-03-29T03:20:48+00:00","tags":[],"pid":1,"method":"post","statusCode":200,"req":{"url":"/elasticsearch/.kibana/index-pattern/logstash-*","method":"post","headers":{"host":"127.0.0.1:8001","connection":"keep-alive","content-length":"2179","accept":"application/json, text/plain, */*","origin":"http://127.0.0.1:8001","kbn-xsrf-token":"92f117c64fc2222e4cc47956944012216edbbec2ac2dea0d607d8bb43086789b","user-agent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/49.0.2623.87 Safari/537.36","content-type":"application/json;charset=UTF-8","referer":"http://127.0.0.1:8001/app/kibana","accept-encoding":"gzip, deflate","accept-language":"en-US,en;q=0.8","cookie":"__utma=96992031.1745009968.1423698390.1423720406.1423720406.1; _ga=GA1.1.1745009968.1423
message: <30>2016-03-29T03:20:48Z swarm-101 docker/5c35d1765a3a[6503]: {"type":"response","@timestamp":"2016-03-29T03:20:48+00:00","tags":[],"pid":1,"method":"post","statusCode":200,"req":{"url":"/elasticsearch/.kibana/config/4.3.0/_update","method":"post","headers":{"host":"127.0.0.1:8001","connection":"keep-alive","content-length":"53","accept":"application/json, text/plain, */*","origin":"http://127.0.0.1:8001","kbn-xsrf-token":"92f117c64fc2222e4cc47956944012216edbbec2ac2dea0d607d8bb43086789b","user-agent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/49.0.2623.87 Safari/537.36","content-type":"application/json;charset=UTF-8","referer":"http://127.0.0.1:8001/app/kibana","accept-encoding":"gzip, deflate","accept-language":"en-US,en;q=0.8","cookie":"__utma=96992031.1745009968.1423698390.1423720406.1423720406.1; _ga=GA1.1.1745009968.142369
message: [2016-03-29 03:20:48,310][INFO][cluster.metadata] [Jessica Jones] [.kibana] update_mapping [config]
@version: 1 @timestamp: March 29th 2016, 14:20:48.487 host: 172.17.8.103 type: syslog tags: _grokparsefailure syslog_severity_code: 5 syslog_facility_code: 1 syslog_facility: user-level syslog_severity: notice _id: AVPAYla58ae1ExblYCcf _type: syslog _index: logstash-2016.03.29 _score:

Chapter 9: Best Practices for the Real World

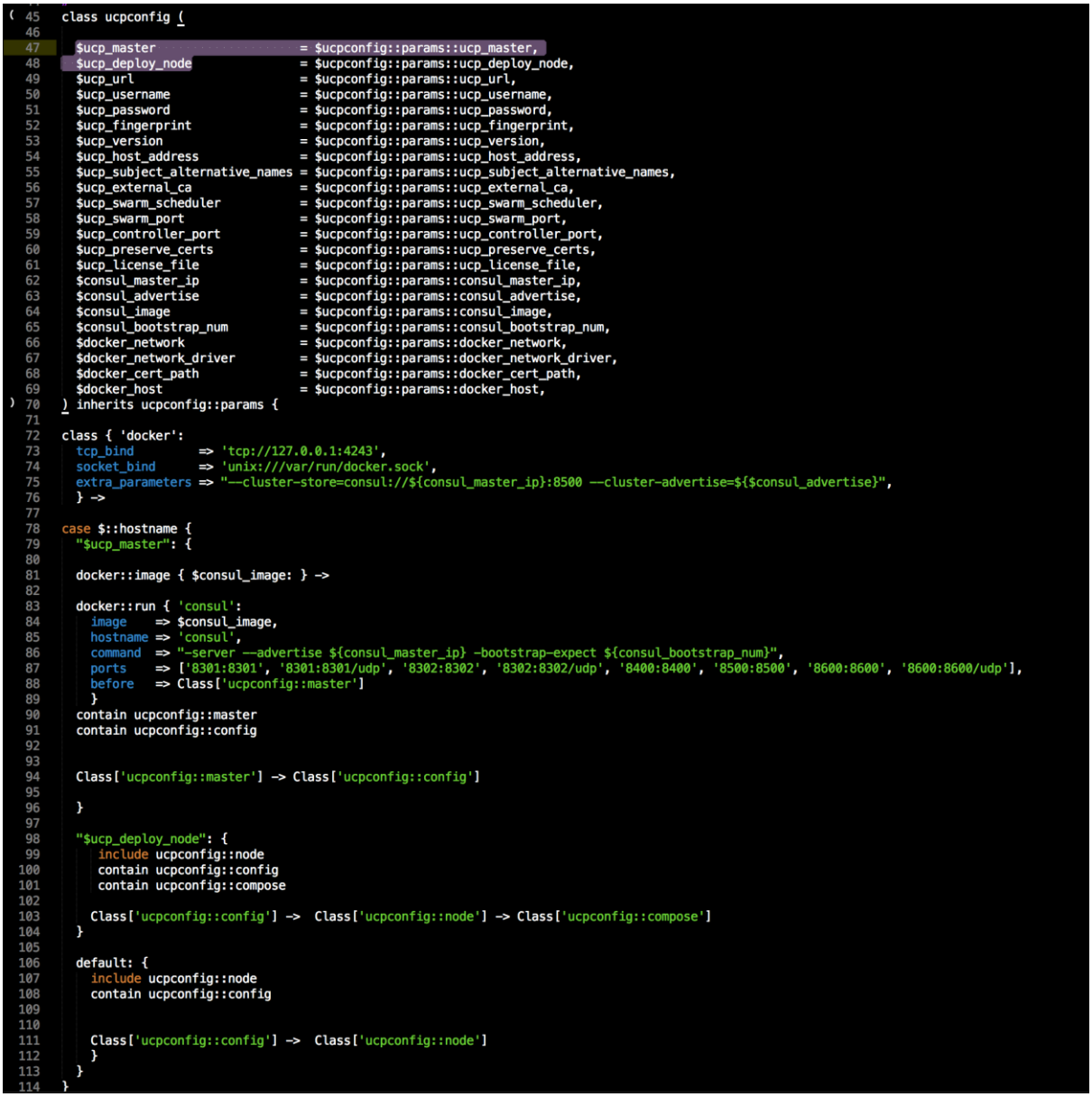


```
---
-
  box: scottyc/ubuntu-14-04-puppet-kernel-4-2
  cpu: 2
  ip: "172.17.10.101"
  name: ucp-01
  forward_ports:
      - { guest: 443, host: 8443 }
  ram: 4096
  shell_commands:
      - { shell: 'apt-get update -y' }
      - { shell: 'apt-get install -y wget git' }
      - { shell: 'mkdir ~/.docker || true && cp /vagrant/config.json ~/.docker/' }
      - { shell: 'mkdir /etc/docker/ || true && cp /vagrant/docker_subscription.lic /etc/docker/subscription.lic' }
      - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
      - { shell: 'cp /home/vagrant/ucp-01/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
      - { shell: 'cp /home/vagrant/ucp-01/modules/* -R /tmp/modules || true' }


-
  box: scottyc/ubuntu-14-04-puppet-kernel-4-2
  cpu: 2
  ip: "172.17.10.102"
  name: ucp-02
  forward_ports:
      - { guest: 443, host: 9443 }
  ram: 4096
  shell_commands:
      - { shell: 'apt-get update -y' }
      - { shell: 'apt-get install -y wget git' }
      - { shell: 'mkdir ~/.docker || true && cp /vagrant/config.json ~/.docker/' }
      - { shell: 'mkdir /etc/docker/ || true && cp /vagrant/docker_subscription.lic /etc/docker/subscription.lic' }
      - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
      - { shell: 'cp /home/vagrant/ucp-02/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
      - { shell: 'cp /home/vagrant/ucp-02/modules/* -R /tmp/modules || true' }

-
  box: scottyc/ubuntu-14-04-puppet-kernel-4-2
  cpu: 2
  ip: "172.17.10.103"
  name: ucp-03
  forward_ports:
      - { guest: 443, host: 10443 }
  ram: 4096
  shell_commands:
      - { shell: 'apt-get update -y' }
      - { shell: 'apt-get install -y wget git' }
      - { shell: 'mkdir ~/.docker || true && cp /vagrant/config.json ~/.docker/' }
      - { shell: 'mkdir /etc/docker/ || true && cp /vagrant/docker_subscription.lic /etc/docker/subscription.lic' }
      - { shell: '/opt/puppet/bin/gem install r10k && ln -s /opt/puppet/bin/r10k /usr/bin/r10k || true'}
      - { shell: 'cp /home/vagrant/ucp-03/Puppetfile /tmp && cd /tmp && r10k puppetfile install --verbose' }
      - { shell: 'cp /home/vagrant/ucp-03/modules/* -R /tmp/modules || true' }
```

```
#!/usr/bin/ruby env

require "socket"
$hostname = Socket.gethostname

forge 'http://forge.puppetlabs.com'

mod 'garethr/docker', :git => 'https://github.com/scotty-c/garethr-docker.git'
mod 'puppetlabs/apt'
mod 'puppetlabs/docker_ucp'
mod 'puppetlabs/stdlib'
mod 'maestrodev/wget'
```

```
class ucpconfig::params {
  $ucp_master                    = ''
  $ucp_deploy_node               = ''
  $ucp_url                       = ''
  $ucp_username                  = ''
  $ucp_password                  = ''
  $ucp_fingerprint               = $::ucp_fingerprint
  $ucp_version                   = '1.0.0'
  $ucp_host_address              = ''
  $ucp_subject_alternative_names = ''
  $ucp_external_ca               = false
  $ucp_swarm_scheduler           = 'binpack'
  $ucp_swarm_port                = ''
  $ucp_controller_port           = '8443'
  $ucp_preserve_certs            = 'true'
  $ucp_license_file              = ''
  $consul_master_ip              = ''
  $consul_advertise              = ''
  $consul_image                  = 'scottyc/consul'
  $consul_bootstrap_num          = '1'
  $docker_network                = 'private-net'
  $docker_network_driver         = 'overlay'
  $docker_cert_path              = ''
  $docker_host                   = ''
}
```

```
Facter.add('ucp_fingerprint') do
  setcode do
    Facter::Core::Execution.exec("echo -n | openssl s_client -connect 172.17.10.101:443 2> /dev/null | sed -ne '/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p' | openssl x509 -noout -fingerprint -sha1 | cut -d= -f2")
  end
end
```



```
class ucpconfig (

  $ucp_master                    = $ucpconfig::params::ucp_master,
  $ucp_deploy_node               = $ucpconfig::params::ucp_deploy_node,
  $ucp_url                       = $ucpconfig::params::ucp_url,
  $ucp_username                  = $ucpconfig::params::ucp_username,
  $ucp_password                  = $ucpconfig::params::ucp_password,
  $ucp_fingerprint               = $ucpconfig::params::ucp_fingerprint,
  $ucp_version                   = $ucpconfig::params::ucp_version,
  $ucp_host_address              = $ucpconfig::params::ucp_host_address,
  $ucp_subject_alternative_names = $ucpconfig::params::ucp_subject_alternative_names,
  $ucp_external_ca               = $ucpconfig::params::ucp_external_ca,
  $ucp_swarm_scheduler           = $ucpconfig::params::ucp_swarm_scheduler,
  $ucp_swarm_port                = $ucpconfig::params::ucp_swarm_port,
  $ucp_controller_port           = $ucpconfig::params::ucp_controller_port,
  $ucp_preserve_certs            = $ucpconfig::params::ucp_preserve_certs,
  $ucp_license_file              = $ucpconfig::params::ucp_license_file,
  $consul_master_ip              = $ucpconfig::params::consul_master_ip,
  $consul_advertise              = $ucpconfig::params::consul_advertise,
  $consul_image                  = $ucpconfig::params::consul_image,
  $consul_bootstrap_num          = $ucpconfig::params::consul_bootstrap_num,
  $docker_network                = $ucpconfig::params::docker_network,
  $docker_network_driver         = $ucpconfig::params::docker_network_driver,
  $docker_cert_path              = $ucpconfig::params::docker_cert_path,
  $docker_host                   = $ucpconfig::params::docker_host,
) inherits ucpconfig::params {

class { 'docker':
  tcp_bind        => 'tcp://127.0.0.1:4243',
  socket_bind     => 'unix:///var/run/docker.sock',
  extra_parameters => "--cluster-store=consul://${consul_master_ip}:8500 --cluster-advertise=${$consul_advertise}",
  } ->

case $::hostname {
  "$ucp_master": {

  docker::image { $consul_image: } ->

  docker::run { 'consul':
    image    => $consul_image,
    hostname => 'consul',
    command  => "-server --advertise ${consul_master_ip} -bootstrap-expect ${consul_bootstrap_num}",
    ports    => ['8301:8301', '8301:8301/udp', '8302:8302', '8302:8302/udp', '8400:8400', '8500:8500', '8600:8600', '8600:8600/udp'],
    before   => Class['ucpconfig::master']
    }
  contain ucpconfig::master
  contain ucpconfig::config


  Class['ucpconfig::master'] -> Class['ucpconfig::config']

  }

  "$ucp_deploy_node": {
     include ucpconfig::node
     contain ucpconfig::config
     contain ucpconfig::compose

    Class['ucpconfig::config'] ->  Class['ucpconfig::node'] -> Class['ucpconfig::compose']
  }

  default: {
    include ucpconfig::node
    contain ucpconfig::config


    Class['ucpconfig::config'] ->  Class['ucpconfig::node']
    }
  }
}
```

```
class ucpconfig::master(

  $ucp_version                  = $ucpconfig::ucp_version,
  $ucp_host_address             = $ucpconfig::ucp_host_address,
  $ucp_subject_alternative_names = $ucpconfig::ucp_subject_alternative_names,
  $ucp_external_ca              = $ucpconfig::ucp_external_ca,
  $ucp_swarm_scheduler          = $ucpconfig::ucp_swarm_scheduler,
  $ucp_swarm_port               = $ucpconfig::ucp_swarm_port,
  $ucp_controller_port          = $ucpconfig::ucp_controller_port,
  $ucp_preserve_certs           = $ucpconfig::ucp_preserve_certs,
  $ucp_license_file             = $ucpconfig::ucp_license_file,

) {

class { 'docker_ucp':
  controller                => true,
  host_address              => $ucp_host_address,
  version                   => $ucp_version,
  usage                     => false,
  tracking                  => false,
  subject_alternative_names => $ucp_subject_alternative_names,
  external_ca               => $ucp_external_ca,
  swarm_scheduler           => $ucp_swarm_scheduler,
  swarm_port                => $ucp_swarm_port,
  controller_port           => $ucp_controller_port,
  preserve_certs            => $ucp_preserve_certs,
  docker_socket_path        => '/var/run/docker.sock',
  license_file              => $ucp_license_file,
  require                   => Class['docker']
  }
}
```

```
class { 'docker_ucp':
  controller                => true,
  host_address              => $ucp_host_address,
  version                   => $ucp_version,
  usage                     => false,
  tracking                  => false,
  subject_alternative_names => $ucp_subject_alternative_names,
  external_ca               => $ucp_external_ca,
  swarm_scheduler           => $ucp_swarm_scheduler,
  swarm_port                => $ucp_swarm_port,
  controller_port           => $ucp_controller_port,
  preserve_certs            => $ucp_preserve_certs,
  docker_socket_path        => '/var/run/docker.sock',
  license_file              => $ucp_license_file,
  require                   => Class['docker']
  }
}
```

```
1  class ucpconfig::node (
2
3    $ucp_url                       = $ucpconfig::ucp_url,
4    $ucp_username                  = $ucpconfig::ucp_username,
5    $ucp_password                  = $ucpconfig::ucp_password,
6    $ucp_fingerprint               = $ucpconfig::ucp_fingerprint,
7    $ucp_version                   = $ucpconfig::ucp_version,
8    $ucp_host_address              = $ucpconfig::ucp_host_address,
9    $ucp_subject_alternative_names = $ucpconfig::ucp_subject_alternative_names,
10
11 ){
12
13 class { 'docker_ucp':
14   ucp_url                  => $ucp_url,
15   fingerprint              => $ucp_fingerprint,
16   username                 => $ucp_username,
17   password                 => $ucp_password ,
18   host_address             => $ucp_host_address,
19   subject_alternative_names => $ucp_subject_alternative_names,
20   replica                  => true,
21   version                  => $ucp_version,
22   usage                    => false,
23   tracking                 => false,
24   require                  => Class['docker']
25   }
26 }
27
```

```
class ucpconfig::config (

 $ucp_url               = $ucpconfig::ucp_url,
 $ucp_username          = $ucpconfig::ucp_username,
 $ucp_password          = $ucpconfig::ucp_password,
 $docker_network        = $ucpconfig::docker_network,
 $docker_network_driver = $ucpconfig::docker_network_driver,
 $docker_cert_path      = $ucpconfig::docker_cert_path,
 $docker_host           = $ucpconfig::docker_host,
 ) {


 package { ['curl', 'zip', 'jq']:
   ensure => installed,
   }

 file { '/etc/docker/get_ca.sh':
   ensure  => file,
   content => template("ucpconfig/get_ca.sh.erb"),
   }

 exec { 'ca_bundle':
   command => 'sh get_ca.sh',
   path    => '/usr/bin:/usr/sbin:/bin:/usr/local/bin',
   cwd     => $docker_cert_path,
   creates => "${$docker_cert_path}/bundle.zip",
   require => File['/etc/docker/get_ca.sh']
   }

 file { '/etc/profile.d/docker.sh':
   ensure  => present,
   content => template('ucpconfig/docker.sh.erb'),
   mode    => '0644',
   }


  docker_network { $docker_network:
    ensure  => present,
    create  => true,
    driver  => $docker_network_driver,
    require => File['/etc/profile.d/docker.sh']
    }
}
```

```
class ucpconfig::config (

 $ucp_url               = $ucpconfig::ucp_url,
 $ucp_username          = $ucpconfig::ucp_username,
 $ucp_password          = $ucpconfig::ucp_password,
 $docker_network        = $ucpconfig::docker_network,
 $docker_network_driver = $ucpconfig::docker_network_driver,
 $docker_cert_path      = $ucpconfig::docker_cert_path,
 $docker_host           = $ucpconfig::docker_host,
 ) {
```

```
 package { ['curl', 'zip', 'jq']:
   ensure => installed,
   }
```

```
 file { '/etc/docker/get_ca.sh':
   ensure  => file,
   content => template("ucpconfig/get_ca.sh.erb"),
   }
```

```
1  #!/bin/bash
2  AUTHTOKEN=$(curl -sk -d '{"username":"<%= @ucp_username %>","password":"<%= @ucp_password %>"}' <%= @ucp_url %>/auth/login | jq -r .auth_token) && \
3  curl -k -H "Authorization: Bearer $AUTHTOKEN" <%= @ucp_url %>/api/clientbundle -o bundle.zip && \
4  unzip bundle.zip
```

```
21
22   exec { 'ca_bundle':
23     command => 'sh get_ca.sh',
24     path    => '/usr/bin:/usr/sbin:/bin:/usr/local/bin',
25     cwd     => $docker_cert_path,
26     creates => "${$docker_cert_path}/ca.pem",
27     require => File['/etc/docker/get_ca.sh']
28     }
```

```
file { '/etc/profile.d/docker.sh':
  ensure  => present,
  content => template('ucpconfig/docker.sh.erb'),
  mode    => '0644',
  }
```

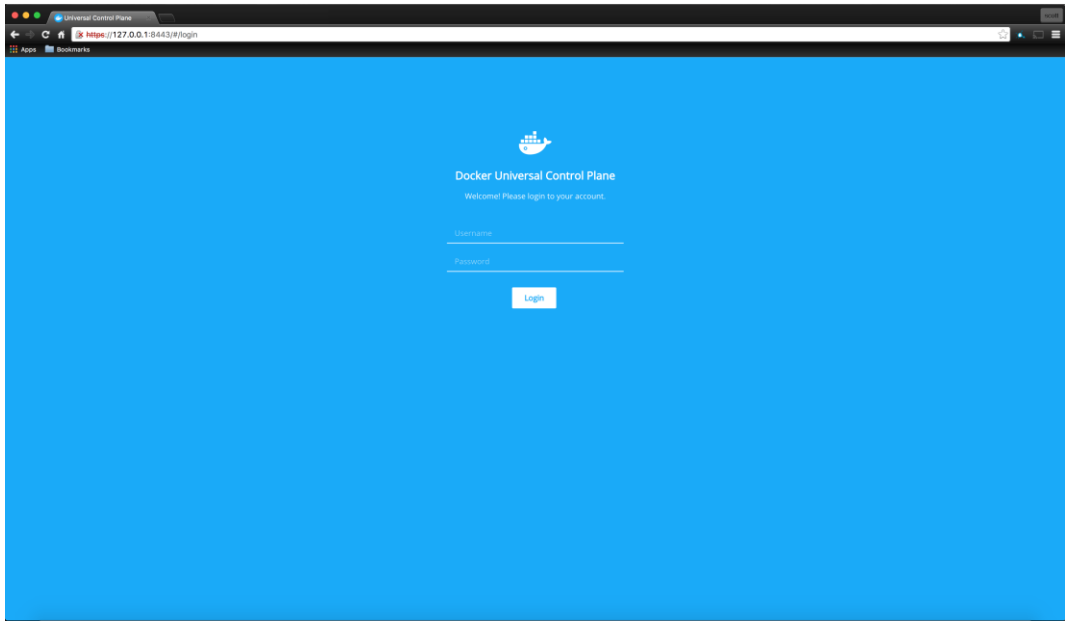
```
1  #!/bin/bash
2  export DOCKER_TLS_VERIFY=1
3  export DOCKER_CERT_PATH=<%= @docker_cert_path %>
4  export DOCKER_HOST=<%= @docker_host %>
5
```

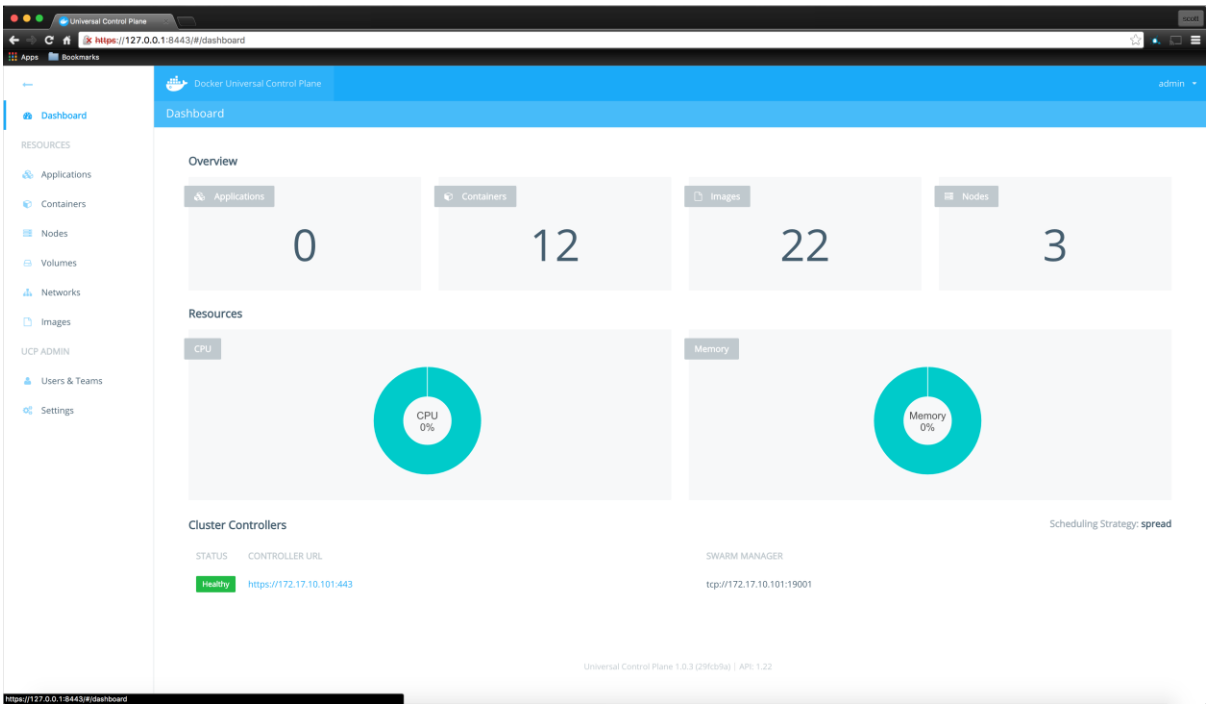
```
  docker_network { $docker_network:
    ensure  => present,
    create  => true,
    driver  => $docker_network_driver,
    require => File['/etc/profile.d/docker.sh']
    }
}
```

```
ucpconfig::ucp_master: ucp-01
ucpconfig::ucp_deploy_node: ucp-03
ucpconfig::ucp_url: https://172.17.10.101
ucpconfig::ucp_username: admin
ucpconfig::ucp_password: orca
ucpconfig::ucp_version: 1.0.3
ucpconfig::ucp_host_address: "%{::ipaddress_eth1}"
ucpconfig::ucp_subject_alternative_names: "%{::ipaddress_eth0}"
ucpconfig::ucp_external_ca: false
ucpconfig::ucp_swarm_scheduler: spread
ucpconfig::ucp_swarm_port: 19001
ucpconfig::ucp_controller_port: 443
ucpconfig::ucp_preserve_certs: true
ucpconfig::ucp_license_file: /etc/docker/subscription.lic
ucpconfig::consul_master_ip: 172.17.10.101
ucpconfig::consul_advertise: eth1:2376
ucpconfig::consul_image: scottyc/consul
ucpconfig::consul_bootstrap_num: 1
ucpconfig::docker_network: swarm-private
ucpconfig::docker_network_drive: overlay
ucpconfig::docker_cert_path: /etc/docker
ucpconfig::docker_host: tcp://172.17.10.101:443

```

```
node 'ucp-01' {
 include ucpconfig
}

node 'ucp-02' {
  include ucpconfig
}

node 'ucp-03' {
  include ucpconfig
}


```

```
==> ucp-03: stdin: is not a tty
==> ucp-03: Running provisioner: puppet...
==> ucp-03: Running Puppet with default.pp...
==> ucp-03: stdin: is not a tty
==> ucp-03: Info: Loading facts
==> ucp-03: Info: Loading facts
==> ucp-03: Info: Loading facts
==> ucp-03: Warning: Scope(Apt::Source[docker]): $include_src is deprecated and will be removed in the next major release, please use $include => { 'src' => false } instead
==> ucp-03: Warning: Scope(Apt::Source[docker]): $required_packages is deprecated and will be removed in the next major release, please use package resources instead.
==> ucp-03: Warning: Scope(Apt::Source[docker]): $key_source is deprecated and will be removed in the next major release, please use $key => { 'source' => https://apt.dockerproject.org/gpg } instead.
==> ucp-03: Warning: Scope(Apt::Key[Add key: 58118E89F3A912897C070ADBF76221572C52609D from Apt::Source docker]): $key_source is deprecated and will be removed in the next major release. Please use $source instead.
==> ucp-03: Notice: Compiled catalog for localhost in environment production in 0.80 seconds
==> ucp-03: Info: Applying configuration version '1460002952'
==> ucp-03: Notice: /Stage[main]/Apt/File[preferences]/ensure: created
==> ucp-03: Info: /Stage[main]/Apt/File[preferences]: Scheduling refresh of Class[Apt::Update]
==> ucp-03: Notice: /Stage[main]/Docker::Compose/Exec[Install Docker Compose 1.6.2]/returns: executed successfully
==> ucp-03: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose-1.6.2]/mode: mode changed '0644' to '0755'
==> ucp-03: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose]/ensure: created
==> ucp-03: Notice: /Stage[main]/Ucpconfig::Config/Package[zip]/ensure: ensure changed 'purged' to 'present'
==> ucp-03: Notice: /Stage[main]/Ucpconfig::Config/Package[jq]/ensure: ensure changed 'purged' to 'present'
==> ucp-03: Notice: /Stage[main]/Ucpconfig::Config/File[/etc/docker/get_ca.sh]/ensure: defined content as '{md5}d9f2643993460b200c4a028f8af9c012'
==> ucp-03: Notice: /Stage[main]/Ucpconfig::Config/Exec[ca_bundle]/returns: executed successfully
==> ucp-03: Notice: /Stage[main]/Ucpconfig::Config/File[/etc/profile.d/docker.sh]/ensure: created
==> ucp-03: Notice: /Stage[main]/Apt/Apt::Setting[conf-update-stamp]/File[/etc/apt/apt.conf.d/15update-stamp]/ensure: defined content as '{md5}0962d70c4ec78bbfa6f3544ae0c41974'
==> ucp-03: Info: /Stage[main]/Apt/Apt::Setting[conf-update-stamp]/File[/etc/apt/apt.conf.d/15update-stamp]: Scheduling refresh of Class[Apt::Update]
==> ucp-03: Notice: /Stage[main]/Docker::Repos/Apt::Source[docker]/Apt::Key[Add key: 58118E89F3A912897C070ADBF76221572C52609D from Apt::Source docker]/Apt_key[Add key: 58118E89F3A912897C070ADBF76221572C52609D from Apt::Source docker]/ensure: created
==> ucp-03: Notice: /Stage[main]/Docker::Repos/Apt::Source[docker]/Apt::Pin[docker]/Apt::Setting[pref-docker]/File[/etc/apt/preferences.d/docker.pref]/ensure: created
==> ucp-03: Notice: /Stage[main]/Docker::Repos/Apt::Source[docker]/Apt::Setting[list-docker]/File[/etc/apt/sources.list.d/docker.list]/ensure: created
==> ucp-03: Info: /Stage[main]/Docker::Repos/Apt::Source[docker]/Apt::Setting[list-docker]/File[/etc/apt/sources.list.d/docker.list]: Scheduling refresh of Class[Apt::Update]
==> ucp-03: Info: Class[Apt::Update]: Scheduling refresh of Exec[apt_update]
==> ucp-03: Notice: /Stage[main]/Apt::Update/Exec[apt_update]: Triggered 'refresh' from 1 events
==> ucp-03: Notice: /Stage[main]/Docker::Repos/Package[cgroup-lite]/ensure: ensure changed 'purged' to 'present'
==> ucp-03: Notice: /Stage[main]/Docker::Install/Package[docker]/ensure: ensure changed 'purged' to 'present'
==> ucp-03: Info: Computing checksum on file /etc/init.d/docker
==> ucp-03: Info: /Stage[main]/Docker::Service/File[/etc/init.d/docker]: Filebucketed /etc/init.d/docker to puppet with sum db8c4e5c7e8311b40b590489eedaa406
==> ucp-03: Notice: /Stage[main]/Docker::Service/File[/etc/init.d/docker]/ensure: ensure changed 'file' to 'link'
==> ucp-03: Info: /Stage[main]/Docker::Service/File[/etc/init.d/docker]: Scheduling refresh of Service[docker]
==> ucp-03: Info: Computing checksum on file /etc/default/docker
==> ucp-03: Info: /Stage[main]/Docker::Service/File[/etc/default/docker]: Filebucketed /etc/default/docker to puppet with sum df04972d0cf6aefc63ad37c69a26a8bb
==> ucp-03: Notice: /Stage[main]/Docker::Service/File[/etc/default/docker]/content: content changed '{md5}df04972d0cf6aefc63ad37c69a26a8bb' to '{md5}f46885ec7d05289abd871d32aae5f020'
==> ucp-03: Info: /Stage[main]/Docker::Service/File[/etc/default/docker]: Scheduling refresh of Service[docker]
==> ucp-03: Notice: /Stage[main]/Docker::Service/Service[docker]: Triggered 'refresh' from 2 events
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: Unable to find image 'docker/ucp:latest' locally
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: latest: Pulling from docker/ucp
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: b7bf206848fc: Pulling fs layer
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: b7bf206848fc: Download complete
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: b7bf206848fc: Pull complete
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: b7bf206848fc: Pull complete
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: Digest: sha256:a7910f80980adb67999539e695471d0d95529d139ecde8f2398141be0a7a195f6
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: Status: Downloaded newer image for docker/ucp:latest
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: time="2016-04-07T04:23:29Z" level=info msg="Pulling required images... (this may take a while)"
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: time="2016-04-07T04:24:23Z" level=warning msg="None of the hostnames we'll be using in the UCP certificates [ucp-03 127.0.0.1 172.18.0.1 172.17.10.103 10.0.2.15] contain a domai
n component.  Your generated certs may fail TLS validation unless you only use one of these shortnames or IPs to connect.  You can use the --san flag to add more aliases"
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: INFO[0000] This engine will join UCP and advertise itself with host address 172.17.10.103 - If this is incorrect, please specify an alternative address with the '--host-address'
 flag
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: INFO[0000] Verifying your system is compatible with UCP
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: INFO[0005] Starting local swarm containers
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: executed successfully
==> ucp-03: Info: checking if docker network exists
==> ucp-03: Notice: Finished catalog run in 116.50 seconds
```

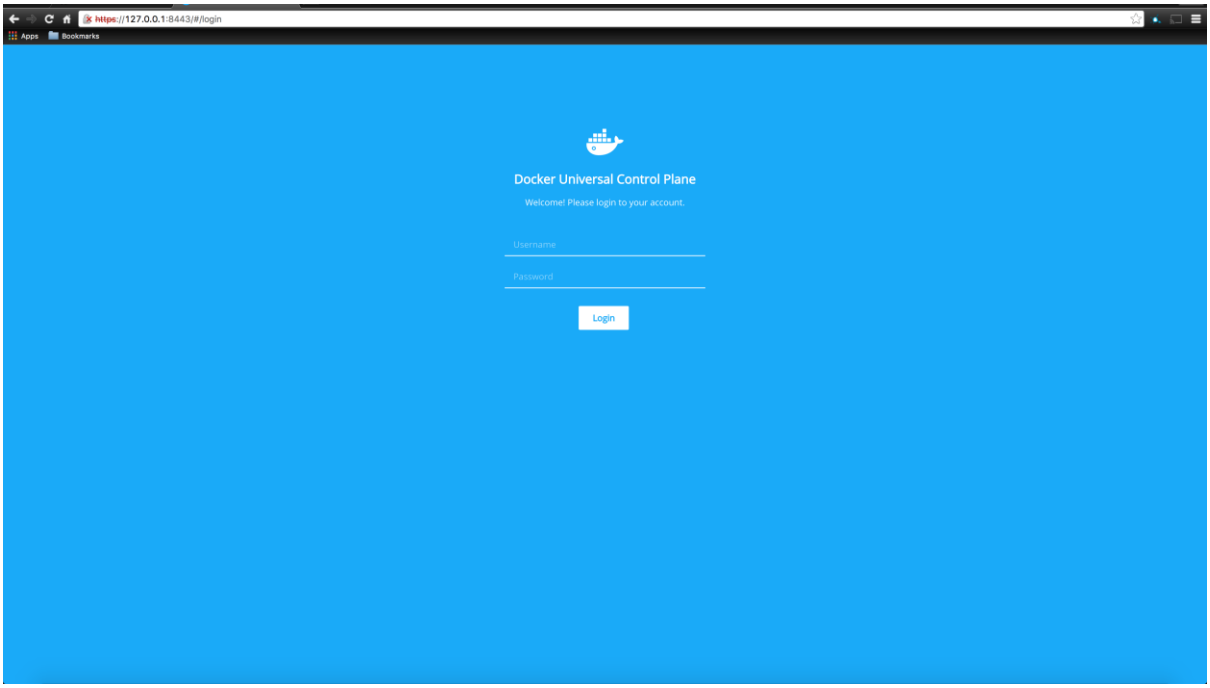


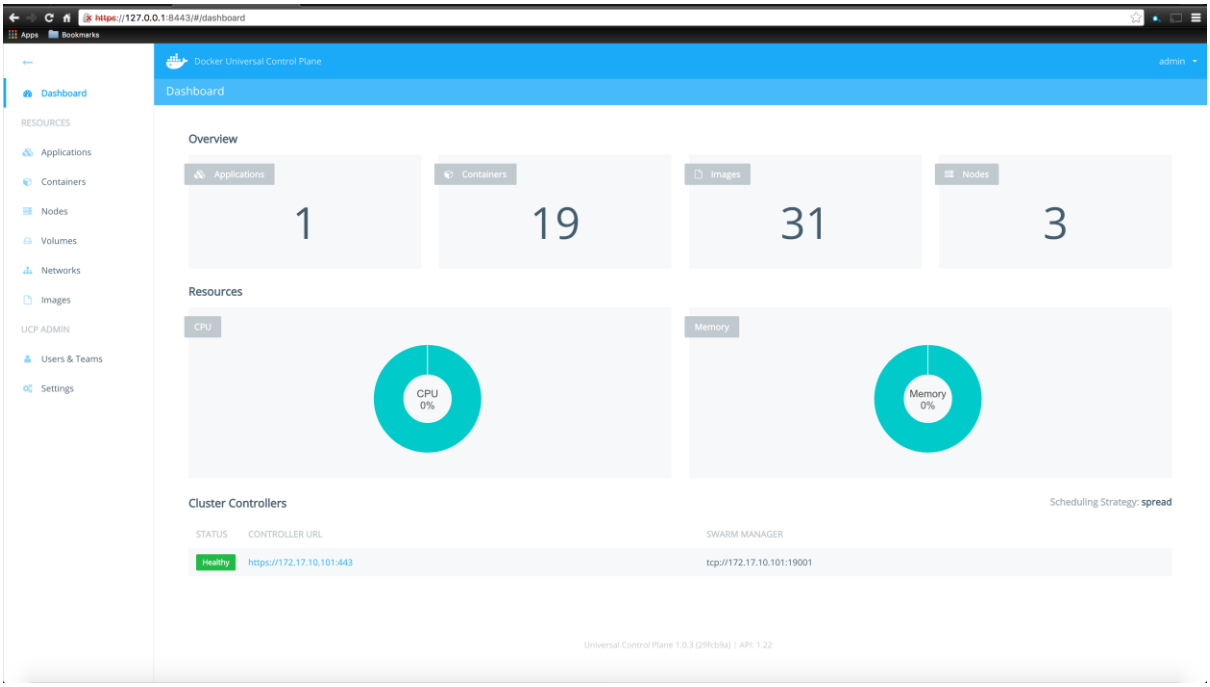


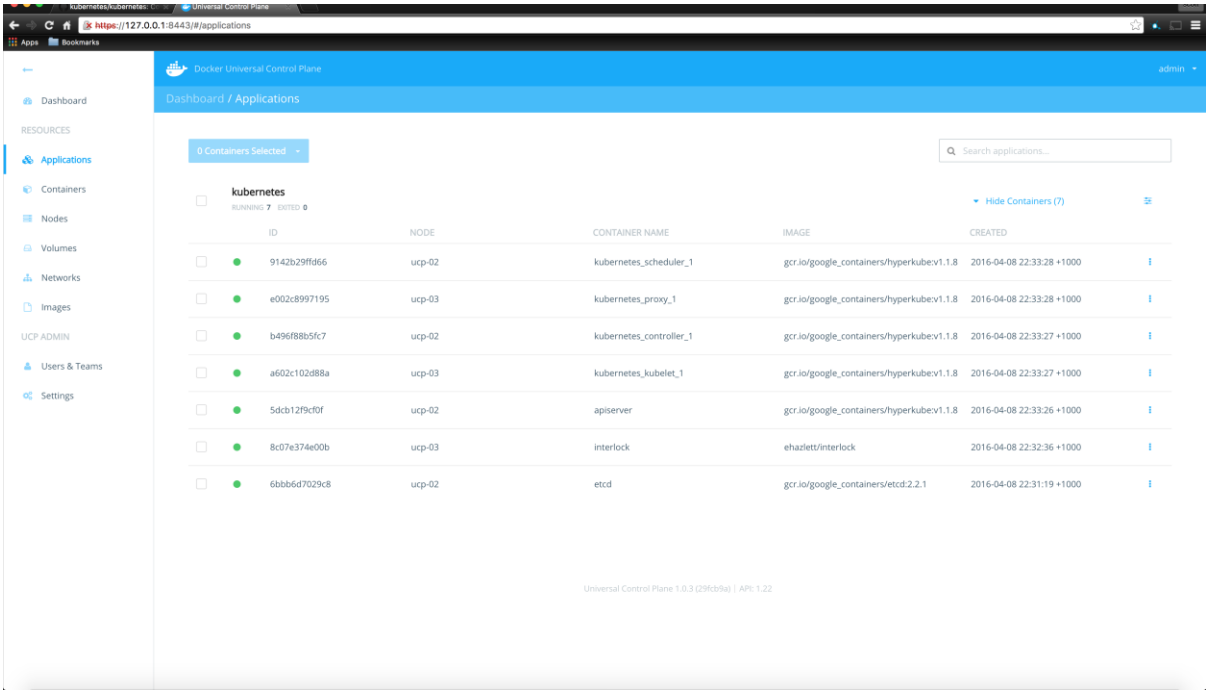

```
class ucpconfig::compose {

 file { '/etc/kubernetes':
   ensure => directory,
   }

 file { '/etc/kubernetes/docker-compose.yml':
   ensure  => file,
   content => template('ucpconfig/kubernertes.yml.erb'),
   require => File['/etc/kubernetes']
   }

  exec { 'docker-compose':
    command => 'bash -l -c "docker-compose -f /etc/kubernetes/docker-compose.yml up -d"',
    path    => '/usr/bin:/usr/sbin:/bin:/usr/local/bin',
    unless  => 'docker inspect -f {{.State.Running}} apiserver',
    }
}
```

```
version: '2'
services:
  interlock:
    image: ehazlett/interlock
    container_name: interlock
    ports:
      - "80:80"
      - "8080:8080"
      - "8443:8443"
    volumes:
      - /etc/docker:/etc/docker
    command: "--swarm-url tcp://172.17.10.101:443 --swarm-tls-ca-cert /etc/docker/ca.pem --swarm-tls-cert /etc/docker/cert.pem --swarm-tls-key /etc/docker/key.pem --plugin haproxy start"
    network_mode: swarm-private
    environment:
      - "constraint:node==ucp-03"

  etcd:
    image: gcr.io/google_containers/etcd:2.2.1
    container_name: etcd
    network_mode: swarm-private
    command: ['/usr/local/bin/etcd', '--bind-addr=0.0.0.0:4001', '--data-dir=/var/etcd/data']

  apiserver:
    image: gcr.io/google_containers/hyperkube:v1.1.8
    container_name: apiserver
    ports:
      - "8080:8080"
    network_mode: swarm-private
    command: ["/hyperkube", "apiserver", "--service-cluster-ip-range=10.17.17.1/24", "--address=0.0.0.0", "--etcd_servers=http://etcd:4001", "--cluster_name=kubernetes", "--v=2"]
    environment:
      - INTERLOCK_DATA={"hostname":"kubernetes","domain":"ucp-demo.local"}

  controller:
    image: gcr.io/google_containers/hyperkube:v1.1.8
    command: ["/hyperkube", "controller-manager", "--address=0.0.0.0", "--master=http://apiserver:8080", "--v=2"]
    network_mode: swarm-private
    environment:
      - "affinity:container==*apiserver*"

  scheduler:
    image: gcr.io/google_containers/hyperkube:v1.1.8
    command: ["/hyperkube", "scheduler", "--address=0.0.0.0", "--master=http://apiserver:8080", "--v=2"]
    network_mode: swarm-private
    environment:
      - "affinity:container==*apiserver*"

  kubelet:
    image: gcr.io/google_containers/hyperkube:v1.1.8
    command: ['/hyperkube', 'kubelet', '--containerized' , '--api_servers=http://apiserver:8080', '--v=2', '--address=0.0.0.0', '--enable_server']
    volumes:
      - /:/rootfs:ro
      - /sys:/sys:ro
      - /dev:/dev
      - /var/run/docker.sock:/var/run/docker.sock
      - /var/lib/docker/:/var/lib/docker:ro
      - /var/lib/kubelet/:/var/lib/kubelet:rw
      - /var/run:/var/run:rw
    privileged: true
    # A kubelet shouldn't run alongside another kubelet - One privileged kubelet per node
    network_mode: swarm-private
    environment:
      - "affinity:container!=*kubelet*"

  proxy:
    image: gcr.io/google_containers/hyperkube:v1.1.8
    command: ['/hyperkube', 'proxy', '--master=http://apiserver:8080', '--v=2']
    privileged: true
    # A proxy should run alongside another kubelet but not alongside another proxy
    network_mode: swarm-private
    environment:
      - "affinity:container==*kubelet*"
```


==> ucp-03: Running provisioner: shell...
ucp-03: Running: inline script
==> ucp-03: stdin: is not a tty
==> ucp-03: Running provisioner: puppet...
==> ucp-03: Running Puppet with default.pp...
==> ucp-03: stdin: is not a tty
==> ucp-03: Info: Loading facts
==> ucp-03: Info: Loading facts
ucp-03: Info: Loading facts
==> ucp-03: Warning: Scope(Apt::Source[docker]): $include_src is deprecated and will be removed in the next major release, please use $include => { 'src' => false } instead
==> ucp-03: Warning: Scope(Apt::Source[docker]): $required_packages is deprecated and will be removed in the next major release, please use package resources instead.
==> ucp-03: Warning: Scope(Apt::Source[docker]): $key_source is deprecated and will be removed in the next major release, please use $key => { 'source' => https://apt.dockerproject.org/gpg } instead.
==> ucp-03: Warning: Scope(Apt::Key[Add key: 58118E89F3A912897C070ADBF76221572C52609D from Apt::Source docker]): $key_source is deprecated and will be removed in the next major release. Please use $source instead.
==> ucp-03: Notice: Compiled catalog for localhost in environment production in 0.75 seconds
==> ucp-03: Info: Applying configuration version '1460118526'
==> ucp-03: Notice: /Stage[main]/Apt/File[preferences]/ensure: created
==> ucp-03: Info: /Stage[main]/Apt/File[preferences]: Scheduling refresh of Class[Apt::Update]
==> ucp-03: Notice: /Stage[main]/Docker::Compose/Exec[Install Docker Compose 1.6.2]/returns: executed successfully
==> ucp-03: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose-1.6.2]/mode: mode changed '0644' to '0755'
ucp-03: Notice: /Stage[main]/Docker::Compose/File[/usr/local/bin/docker-compose]/ensure: created
==> ucp-03: Notice: /Stage[main]/Ucpconfig::Config/Package[zip]/ensure: ensure changed 'purged' to 'present'
==> ucp-03: Notice: /Stage[main]/Ucpconfig::Config/Package[jq]/ensure: ensure changed 'purged' to 'present'
==> ucp-03: Notice: /Stage[main]/Ucpconfig::Config/File[/etc/docker/get_ca.sh]/ensure: defined content as '{md5}d9f2643993460b200c4a028f8af9c012'
==> ucp-03: Notice: /Stage[main]/Ucpconfig::Config/Exec[ca_bundle]/returns: executed successfully
==> ucp-03: Notice: /Stage[main]/Ucpconfig::Config/File[/etc/profile.d/docker.sh]/ensure: created
==> ucp-03: Notice: /Stage[main]/Apt/Apt::Setting[conf-update-stamp]/File[/etc/apt/apt.conf.d/15update-stamp]/ensure: defined content as '{md5}0962d70c4ec78bbf6f3544ae0c41974'
==> ucp-03: Info: /Stage[main]/Apt/Apt::Setting[conf-update-stamp]/File[/etc/apt/apt.conf.d/15update-stamp]: Scheduling refresh of Class[Apt::Update]
==> ucp-03: Notice: /Stage[main]/Docker::Repos/Apt::Source[docker]/Apt::Key[Add key: 58118E89F3A912897C070ADBF76221572C52609D from Apt::Source docker]/Apt_key[Add key: 58118E89F3A912897C070ADBF76221572C52609D from Apt::Source docker]/ensure: created
==> ucp-03: Notice: /Stage[main]/Docker::Repos/Apt::Source[docker]/Apt::Pin[docker]/Apt::Setting[pref-docker]/File[/etc/apt/preferences.d/docker.pref]/ensure: created
==> ucp-03: Notice: /Stage[main]/Docker::Repos/Apt::Source[docker]/Apt::Setting[list-docker]/File[/etc/apt/sources.list.d/docker.list]/ensure: created
==> ucp-03: Info: /Stage[main]/Docker::Repos/Apt::Source[docker]/Apt::Setting[list-docker]/File[/etc/apt/sources.list.d/docker.list]: Scheduling refresh of Class[Apt::Update]
==> ucp-03: Info: Class[Apt::Update]: Scheduling refresh of Exec[apt_update]
==> ucp-03: Notice: /Stage[main]/Apt::Update/Exec[apt_update]: Triggered 'refresh' from 1 events
==> ucp-03: Notice: /Stage[main]/Docker::Repos/Package[cgroup-lite]/ensure: ensure changed 'purged' to 'present'
==> ucp-03: Notice: /Stage[main]/Docker::Install/Package[docker]/ensure: ensure changed 'purged' to 'present'
==> ucp-03: Info: Computing checksum on file /etc/init.d/docker
==> ucp-03: Info: /Stage[main]/Docker::Service/File[/etc/init.d/docker]: Filebucketed /etc/init.d/docker to puppet with sum db8c4e5c7e8911b40b590489eedaa406
==> ucp-03: Notice: /Stage[main]/Docker::Service/File[/etc/init.d/docker]/ensure: ensure changed 'file' to 'link'
==> ucp-03: Info: /Stage[main]/Docker::Service/File[/etc/init.d/docker]: Scheduling refresh of Service[docker]
==> ucp-03: Info: Computing checksum on file /etc/default/docker
==> ucp-03: Info: /Stage[main]/Docker::Service/File[/etc/default/docker]: Filebucketed /etc/default/docker to puppet with sum df04972d0cf6aefcf63ad37c69a26a8bb
==> ucp-03: Notice: /Stage[main]/Docker::Service/File[/etc/default/docker]/content: content changed '{md5}df04972d0cf6aefcf63ad37c69a26a8bb' to '{md5}f46885ec7d05289abd871d32aae5f020'
==> ucp-03: Info: /Stage[main]/Docker::Service/File[/etc/default/docker]: Scheduling refresh of Service[docker]
==> ucp-03: Notice: /Stage[main]/Docker::Service/Service[docker]: Triggered 'refresh' from 2 events
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: Unable to find image 'docker/ucp:latest' locally
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: latest: Pulling from docker/ucp
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: b7bf206848fc: Pulling fs layer
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: b7bf206848fc: Verifying Checksum
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: b7bf206848fc: Download complete
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: b7bf206848fc: Pull complete
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: b7bf206848fc: Pull complete
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: Digest: sha256:a7910f8098bdb67999539e695471d0d95529d139ecde8f2398141be0a7a195f6
ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: Status: Downloaded newer image for docker/ucp:latest
ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: time="2016-04-08T12:29:51Z" level=info msg="Pulling required images... (this may take a while)"
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: time="2016-04-08T12:30:46Z" level=warning msg="None of the hostnames we'll be using in the UCP certificates [ucp-03 127.0.0.1 172.18.0.1 172.17.10.103 10.0.2.15] contain a domain component. Your generated certs may fail TLS validation unless you only use one of these shortnames or IPs to connect. You can use the --san flag to add more aliases"
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: INFO[0000] This engine will join UCP and advertise itself with host address 172.17.10.103 - If this is incorrect, please specify an alternative address With the '--host-address' flag
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: INFO[0000] Verifying your system is compatible with UCP
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: INFO[0007] Starting local swarm containers
==> ucp-03: Notice: /Stage[main]/Docker_ucp/Exec[Join Docker Universal Control Plane]/returns: executed successfully
==> ucp-03: Info: checking if docker network exists
==> ucp-03: Notice: /Stage[main]/Ucpconfig::Compose/File[/etc/kubernetes]/ensure: created
==> ucp-03: Notice: /Stage[main]/Ucpconfig::Compose/File[/etc/kubernetes/docker-compose.yml]/ensure: defined content as '{md5}e243a65affc872d11436a93603f369cc'
==> ucp-03: Notice: /Stage[main]/Ucpconfig::Compose/Exec[docker-compose]/returns: executed successfully
==> ucp-03: Notice: Finished catalog run in 282.38 seconds


https://127.0.0.1:8443/#/login
Apps
Bookmarks
Docker Universal Control Plane
Welcome! Please login to your account.
Username
Password
Login


https://127.0.0.1:8443/#/dashboard
Apps
Bookmarks
Docker Universal Control Plane
admin
Dashboard
RESOURCES
Applications
Containers
Nodes
Volumes
Networks
Images
UCP ADMIN
Users & Teams
Settings
Dashboard
Overview
Applications
1
Containers
19
Images
31
Nodes
3
Resources
CPU
CPU
0%
Memory
Memory
0%
Cluster Controllers
Scheduling Strategy: spread
STATUS
CONTROLLER URL
SWARM MANAGER
Healthy
https://172.17.10.101:443
tcp://172.17.10.101:19001
Universal Control Plane 1.0.3 (29fcb9a) | API: 1.22


https://127.0.0.1:8443/#/applications
Docker Universal Control Plane
admin
Dashboard / Applications
Dashboard
RESOURCES
Applications
Containers
Nodes
Volumes
Networks
Images
UCP ADMIN
Users & Teams
Settings
0 Containers Selected
Search applications...
kubernetes
RUNNING 7 EXITED 0
Hide Containers (7)
ID NODE CONTAINER NAME IMAGE CREATED
9142b29ffd66 ucp-02 kubernetes_scheduler_1 gcr.io/google_containers/hyperkube:v1.1.8 2016-04-08 22:33:28 +1000
e002c8997195 ucp-03 kubernetes_proxy_1 gcr.io/google_containers/hyperkube:v1.1.8 2016-04-08 22:33:28 +1000
b496f88b5fc7 ucp-02 kubernetes_controller_1 gcr.io/google_containers/hyperkube:v1.1.8 2016-04-08 22:33:27 +1000
a602c102d88a ucp-03 kubernetes_kubelet_1 gcr.io/google_containers/hyperkube:v1.1.8 2016-04-08 22:33:27 +1000
5dcb12f9cf0f ucp-02 apiserver gcr.io/google_containers/hyperkube:v1.1.8 2016-04-08 22:33:26 +1000
8c07e374e00b ucp-03 interlock ehazlett/interlock 2016-04-08 22:32:36 +1000
6bbb6d7029c8 ucp-02 etcd gcr.io/google_containers/etcd:2.2.1 2016-04-08 22:31:19 +1000
Universal Control Plane 1.0.3 (29fcb9a) | API: 1.22


root@ucp-03:~# wget https://storage.googleapis.com/kubernetes-release/release/v1.1.8/bin/linux/amd64/kubectl
--2016-04-08 05:47:19-- https://storage.googleapis.com/kubernetes-release/release/v1.1.8/bin/linux/amd64/kubectl
Resolving storage.googleapis.com (storage.googleapis.com)... 216.58.199.80, 2404:6800:4006:801::2010
Connecting to storage.googleapis.com (storage.googleapis.com)|216.58.199.80|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 22589688 (22M) [application/octet-stream]
Saving to: 'kubectl'
100%[=>] 22,589,688 2.29MB/s in 10s
2016-04-08 05:47:30 (2.08 MB/s) - 'kubectl' saved [22589688/22589688]
root@ucp-03:~#


root@ucp-03:~# ./kubectl -s kubernetes.ucp-demo.local get nodes
NAME LABELS STATUS AGE
a602c102d88a kubernetes.io/hostname=a602c102d88a Ready 26m
root@ucp-03:~#