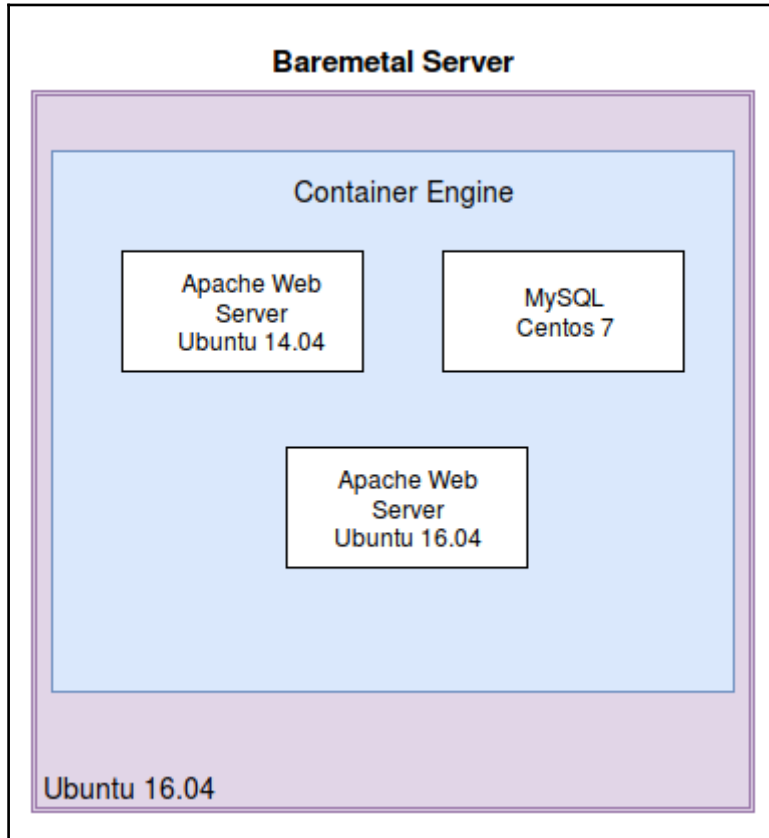
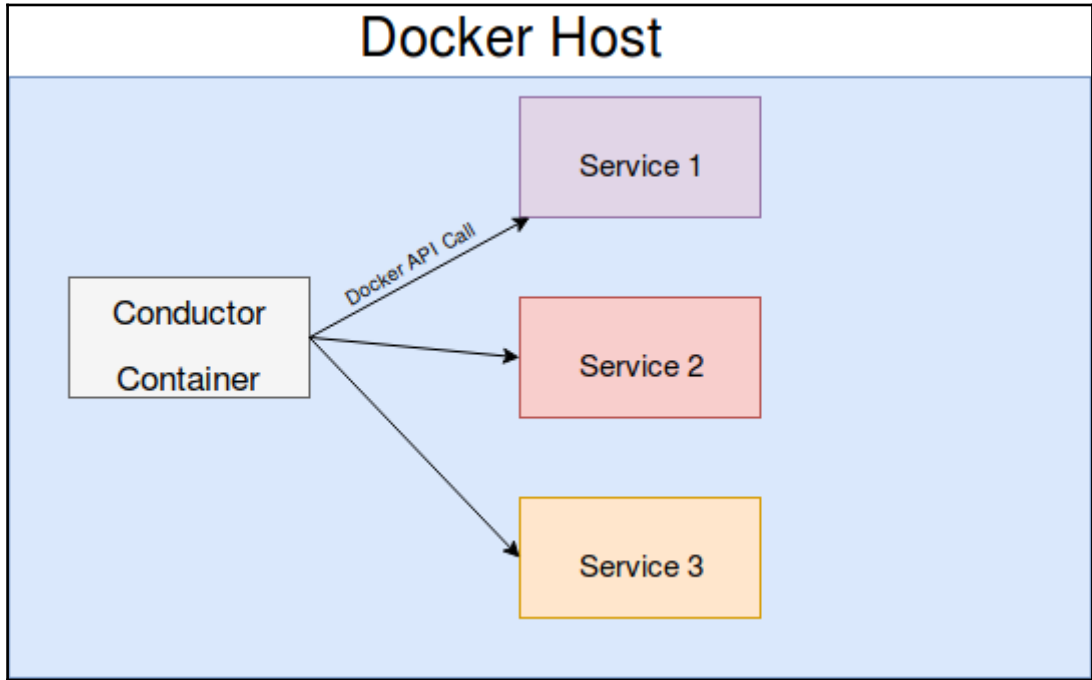


Chapter 1: Building Containers with Docker



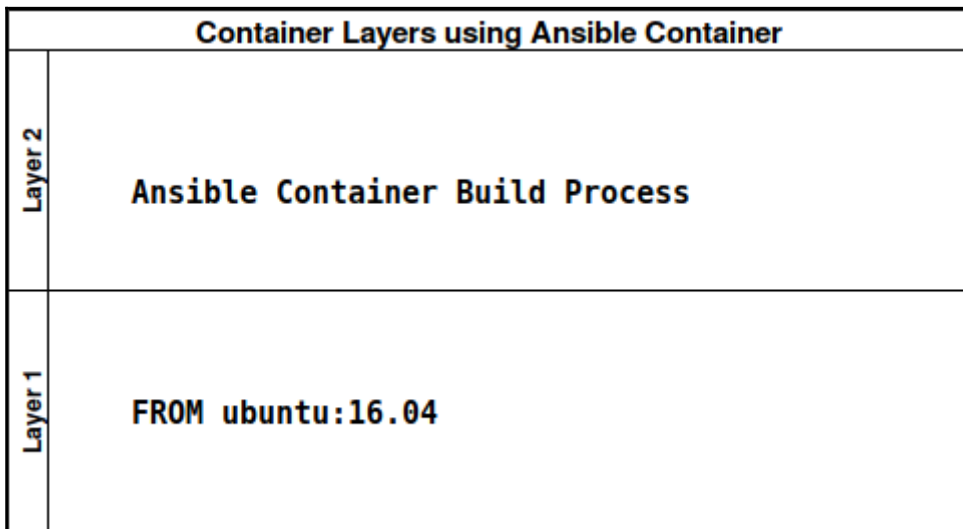
Chapter 2: Working with Ansible Container



Chapter 3: Your First Ansible Container Project

The screenshot shows the Ansible Galaxy interface. At the top, there's a navigation bar with 'ABOUT', 'EXPLORE', 'BROWSE ROLES', 'BROWSE AUTHORS', and 'SIGN IN'. Below this is a search bar with 'Role Type' set to 'Container Enabled' and 'Container App'. The main content area displays two role cards: 'mysql' (1813 stars) and 'nginx' (1557 stars). Both roles are by 'bennojoy' and support 'Enterprise_Linux, Fedora, Ubuntu'. The 'mysql' role has tags 'database, sql' and the 'nginx' role has tags 'web'. A 'POPULAR TAGS' sidebar on the right lists various tags with their respective counts.

Tag	Count
system	4321
development	2237
web	1912
monitoring	903
networking	773
database	732
cloud	666
packaging	621
docker	383
ubuntu	370



Chapter 5: Containers at Scale with Kubernetes

Select

Search projects and folders

Recent All

Name	ID
✓ AC-Kubernetes-Demo	ac-kubernetes-demo

- Container Engine > Container clusters
- Cloud Functions
- Workloads
- Discovery & load balancing
- Configuration
- Storage

STORAGE

Container clusters

<input type="checkbox"/> Name ^	Zone	Cluster size	Total cores	Total memory	Node version	Labels
<input checked="" type="checkbox"/> cluster-1	us-central1-a	3	3 vCPUs	11.25 GB	1.6.9	Connect

Connect to the cluster

Configure `kubectl` command line access by running the following command:

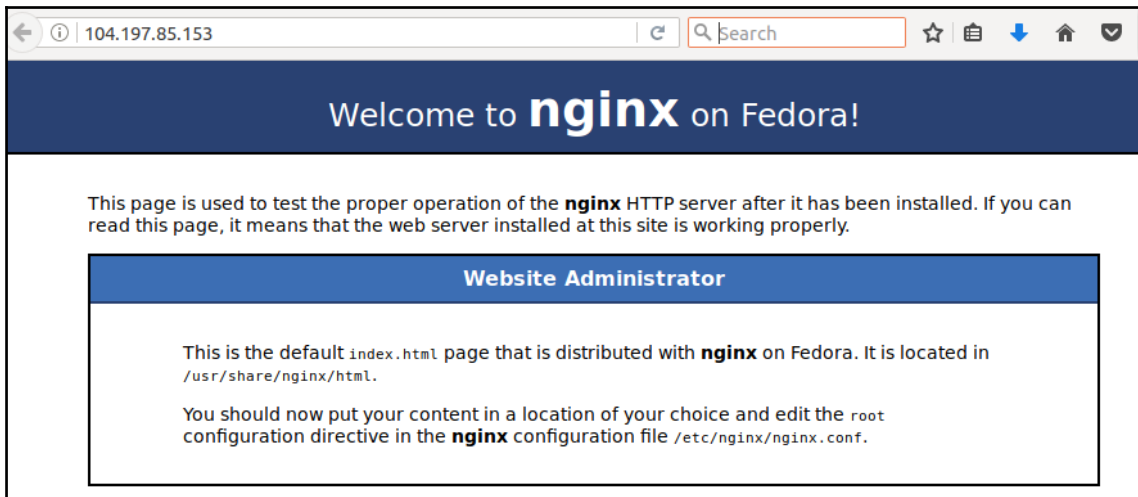
```
$ gcloud container clusters get-credentials cluster-1 --zone us-central1-a --project ac-kubernetes-demo
```

Then start a proxy to connect to the Kubernetes control plane:

```
$ kubectl proxy
```

Then open the Dashboard interface by navigating to the following location in your browser:

<http://localhost:8001/ui>



The screenshot shows a web browser window with the address bar displaying `104.197.85.153`. The page content is as follows:

Welcome to **nginx** on Fedora!

This page is used to test the proper operation of the **nginx** HTTP server after it has been installed. If you can read this page, it means that the web server installed at this site is working properly.

Website Administrator

This is the default `index.html` page that is distributed with **nginx** on Fedora. It is located in `/usr/share/nginx/html`.

You should now put your content in a location of your choice and edit the `root` configuration directive in the **nginx** configuration file `/etc/nginx/nginx.conf`.

Chapter 6: Managing Containers with OpenShift

OPENSIFT ORIGIN

Username

Password

Welcome to OpenShift Origin.

[Log In](#)

Project My Project ▼ [Add to Project](#) ▼ 🔔 ▼ 👤 developer ▼

Overview

Applications >

Builds >

Resources >

Storage

Monitoring

Get started with your project.

Use your source or an example repository to build an application image, or add components like databases and message queues.

[Add to Project](#)

The screenshot shows the OpenShift console interface for a project named "My Project". On the left is a dark sidebar with navigation options: "Overview", "Applications", and "Builds". The main content area has a dark header with a home icon, the project name "Project My Project", and an "Add to Project" dropdown menu. The dropdown menu is open, showing three options: "Browse Catalog" (highlighted), "Deploy Image", and "Import YAML / JSON". Below the menu, the main content area contains the text "Get started with your project." and "Use your source or an example repository to build an application image, or add components like databases and message queues." A blue "Add to Project" button is located at the bottom right of the main content area.

The screenshot shows the details for a builder image named "Python". It features the Python logo and the text "Python". Below this, it says "BUILDS SOURCE CODE". The main description reads: "Build and run Python 3.5 applications on CentOS 7. For more information about using this builder image, including OpenShift considerations, see <https://github.com/sclorg/s2i-python-container/blob/master...>". Underneath, there is a "Version" label and a dropdown menu currently set to "3.5 — latest". At the bottom right, there is a "Select" button.

My Project » Add to Project » Catalog » Python



Python

Build and run Python 3.5 applications on CentOS 7. For more information about using this builder image, including OpenShift considerations, see <https://github.com/sclorg/s2l-python-container/blob/master/3.5/README.md>.

Version: 3.5

*** Name**

Identifies the resources created for this application.

*** Git Repository URL**

Sample repository for python: <https://github.com/openshift/django-ex.git> [Try It ↕](#)

Show [advanced options](#) for source, routes, builds, and deployments.

Create

Cancel



Created application oc-test-deployment in project My Project. [Show Details](#) | [Dismiss](#)

APPLICATION

oc-test-deployment

<http://oc-test-deployment-myproject.192.168.99.100.nip.io>



DEPLOYMENT

oc-test-deployment,
#1



1 pod



oc-test-deployment-myproject.192.168.99.100.nip.io | Search

Welcome to your Django application on OpenShift

How to use this example application

For instructions on how to use this application with OpenShift, start by reading the [Developer Guide](#).

Deploying code changes

The source code for this application is available to be forked from the [OpenShift GitHub repository](#). You can configure a webhook in your repository to make OpenShift automatically start a build whenever you push your code:

1. From the Web Console homepage, navigate to your project

Managing your application

Documentation on how to manage your application Console or Command Line is available at the [Developer](#)

Web Console

You can use the Web Console to view the state of components and launch new builds.

Command Line

With the [OpenShift command line interface \(CLI\)](#), applications and manage projects from a terminal.

Overview

Applications

Builds

Resources

Deployments [Learn More](#)

Filter by label Add

Name	Last Version	Status	Created	Trigger
oc-test-deployment	#2	Active, 1 replica	2 minutes ago	Manual

Overview

Applications

Builds

Resources

Deployments » oc-test-deployment

oc-test-deployment created a day ago

app oc-test-deployment

History Configuration Environment Events

Container oc-test-deployment Environment Variables

Name	Value
<input type="text" value="Name"/>	<input type="text" value="Value"/>

[Add Environment Variable](#) | [Add Environment Variable Using a Config Map or Secret](#)

Deploy Actions

- Edit
- Pause Rollouts
- Add Storage
- Add Autoscaler
- Edit Resource Limits
- Edit Health Checks
- Edit YAML
- Delete

Overview

Pods [Learn More](#)

Filter by label Add

Name	Status	Containers Ready	Container Restarts	Age
oc-test-deployment-1-11818	Running	1/1	0	4 minutes
oc-test-deployment-1-build	Completed	0/1	0	5 minutes

Applications

Builds

Overview

Pods » [oc-test-deployment-1-11818](#)

oc-test-deployment-1-11818 created 5 minutes ago Actions

[app](#) [oc-test-deployment](#) [deployment](#) [oc-test-deployment-1](#) [deploymentconfig](#) [oc-test-deployment](#)

[Details](#) [Environment](#) [Logs](#) [Terminal](#) [Events](#)

Status

Status: Running

Deployment: [oc-test-deployment, #1](#)

IP: 172.17.0.3

Node: localhost (10.0.2.15)

Restart Policy: Always

Container oc-test-deployment

State: Running since Oct 13, 2017 10:55:50 PM

Ready: true

Restart Count: 0

Applications

Builds

Resources

Storage

Are you sure you want to delete the deployment config '**oc-test-deployment**'?

This will delete the deployment config, all rollout history, and any running pods. **It cannot be undone.** Make sure this is something you really want to do!

Delete

Cancel

← → ⓘ awesomewebapp.192.168.99.100.nip.io | 🔄 🔍 Search ☆ 📅 ⬇

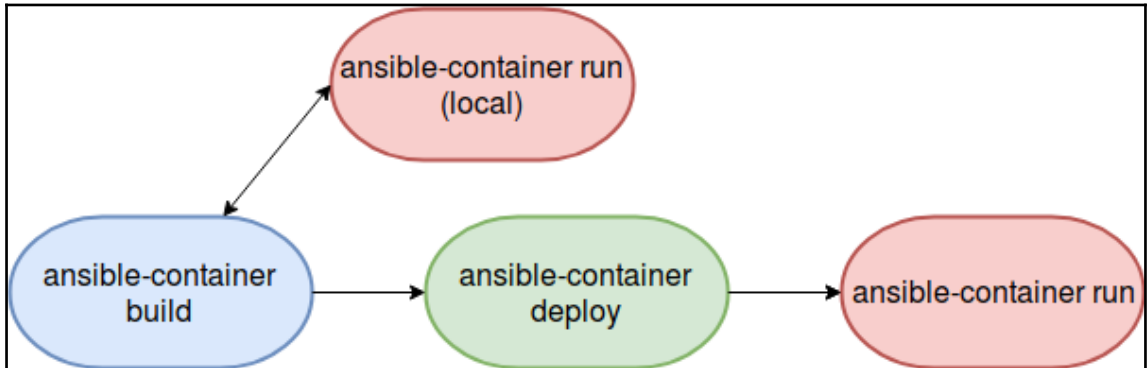
Welcome to nginx!

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.

Chapter 7: Deploying Your First Project



OPENSIFT ORIGIN

Username Welcome to OpenShift Origin.

Password

[Log In](#)

My Projects Sort by

database	created by developer 3 minutes ago	⋮
My Project	Initial developer project	⋮

Project **database** ▼ Add to Project ▼ 🔔 ▼ 👤 developer ▼

Overview

Applications >

Builds >

Resources >


Storage

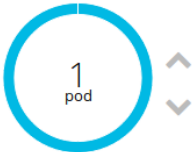
Monitoring

APPLICATION **database** <http://mariadb-database-3306-database.192.168.99.101.nip.io>

DEPLOYMENT **mariadb-database, #1** ⋮

CONTAINER: MARIADB-DATABASE

 **Image:** aric49/mariadb-k8s-mariadb-database:openshift
Ports: 3306/TCP



Networking

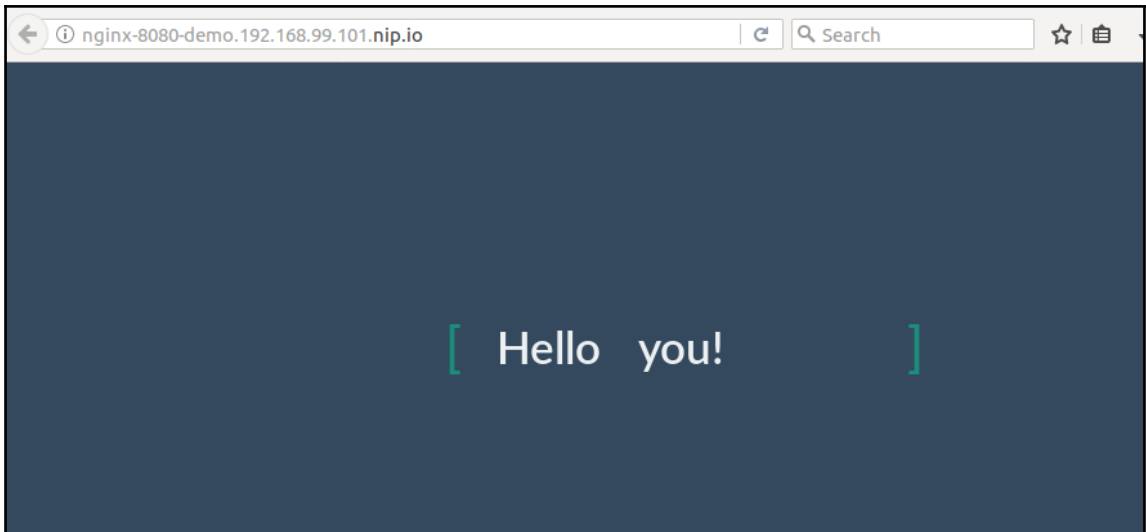
<p>SERVICE Internal Traffic</p> <p>mariadb-database</p> <p>3306/TCP (port-3306-tcp) → 3306</p>	<p>ROUTES External Traffic</p> <p>http://mariadb-database-3306-database.192.168.99.101.nip.io</p> <p>Route mariadb-database-3306, target port port-3306-tcp</p>
---	---

Chapter 8: Building and Deploying a Multi-Container Project

The screenshot shows the Galaxy web interface. At the top left is the Galaxy logo. A search bar contains the text "Choose from option list". A dropdown menu is open, showing "Container Enabled" and "Container App" as options. Below the search bar, there are two role cards: "mysql" with 2175 roles and "nginx" with 1831 roles. To the right, a "POPULAR TAGS" section lists "system" (4569), "development" (2360), and "web" (1997). A "BROWSE ROLES" link is visible in the top right.

The screenshot shows the "My Projects" page. It features a search bar labeled "Filter by keyword", a "Sort by" dropdown set to "Display Name", and a "Create Project" button. Two project cards are listed: "Ansible Container Demo" (created 2 minutes ago) and "My Project" (created an hour ago). Each card includes a description and a vertical ellipsis menu icon.

The screenshot shows the application deployment details for "demo". The URL is "http://nginx-8080-demo.192.168.99.101.nip.io". There are three deployment entries: "django, #1" (1 pod), "nginx, #1" (1 pod), and "postgresql, #2" (Rolling deployment is running... View Events | Cancel). Each entry has a vertical ellipsis menu icon.



[Persistent Volume Claims](#) » **postgres-data**

postgres-data created 22 minutes ago Actions ▾

[Details](#) [Events](#)

Status:	✔ Bound to volume pv0064
Capacity:	allocated 100 GiB
Requested Capacity:	3 GiB
Access Modes:	RWX (Read-Write-Many)

Chapter 9: Going Further with Ansible Container

Built for developers

GitHub is a development platform inspired by the way you work. From **open source** to **business**, you can host and review code, manage projects, and build software alongside millions of other developers.

Username

Email

Password

Use at least one letter, one numeral, and seven characters.

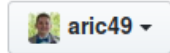
[Sign up for GitHub](#)

By clicking "Sign up for GitHub", you agree to our [terms of service](#) and [privacy policy](#). We'll occasionally send you account related emails.

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner



aric49 ▾

/

Repository name

AwesomeApplication



Great repository names are short and memorable. Need inspiration? How about **crispy-goggles**.

Description (optional)

A repository for an awesome containerized application



Public

Anyone can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.



Initialize this repository with a README

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** ▾

Add a license: **None** ▾



Create repository

Quick setup — if you've done this kind of thing before

or

HTTPS

SSH

https://github.com/aric49/AwesomeApplication.git



We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).