Get Started, Part 6: Deploy your app

Estimated reading time: 7 minutes

Prerequisites

- Install Docker (https://docs.docker.com/install/).
- Get Docker Compose (https://docs.docker.com/compose/overview/) as described in Part 3 prerequisites (https://docs.docker.com/get-started/part3/#prerequisites).
- Get Docker Machine (https://docs.docker.com/machine/overview/) as described in Part 4 prerequisites (https://docs.docker.com/get-started/part4/#prerequisites).
- Read the orientation in Part 1 (https://docs.docker.com/get-started/).
- Learn how to create containers in Part 2 (https://docs.docker.com/get-started/part2/).
- Make sure you have published the `friendlyhello` image you created by pushing it to a registry (https://docs.docker.com/get-started/part2/#share-your-image). We use that shared image here.
- Be sure your image works as a deployed container. Run this command, slotting in your info for `username` , `repo` , and `tag` : `docker run -p 80:80 username/repo:tag` , then visit `http://localhost/` .
- Have the final version of `docker-compose.yml` from Part 5 (https://docs.docker.com/get-started/part5/#persist-the-data) handy.

Introduction

You’ve been editing the same Compose file for this entire tutorial. Well, we have good news. That Compose file works just as well in production as it does on your machine. In this section, we will go through some options for running your Dockerized application.

Choose an option

Docker Enterprise (/get-started/part6/#enterprise)

Docker Engine - Community (/get-started/part6/#community)

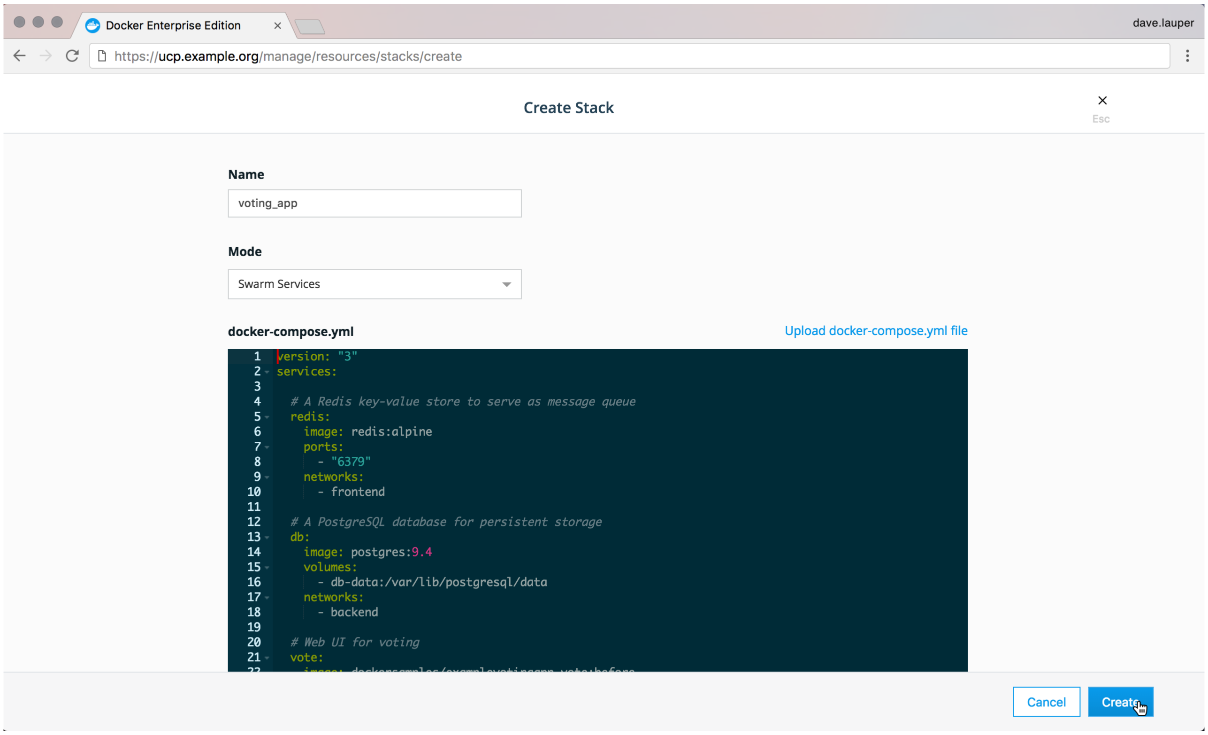
Customers of Docker Enterprise Edition run a stable, commercially-supported version of Docker Engine, and as an add-on they get our first-class management software, Docker Datacenter. You can manage every aspect of your application through the interface using Universal Control Plane, run a private image registry with Docker Trusted Registry, integrate with your LDAP provider, sign production images with Docker Content Trust, and many other features.

Bringing your own server to Docker Enterprise and setting up Docker Datacenter essentially involves two steps:

1. Get Docker Enterprise for your server’s OS from Docker Hub (https://hub.docker.com/search?offering=enterprise&type=edition).
2. Follow the instructions to install Docker Enterprise on your own host (https://docs.docker.com/datacenter/install/linux/).

> **Note**: Running Windows containers? View our Windows Server setup guide (https://docs.docker.com/install/windows/docker-ee/).

Once you’re all set up and Docker Enterprise is running, you can deploy your Compose file from directly within the UI (https://docs.docker.com/ee/ucp/swarm/deploy-multi-service-app/).


Docker Enterprise Edition
dave.lauper
https://ucp.example.org/manage/resources/stacks/create
Create Stack
Esc

Name: voting_app

Mode: Swarm Services

docker-compose.yml

Upload docker-compose.yml file

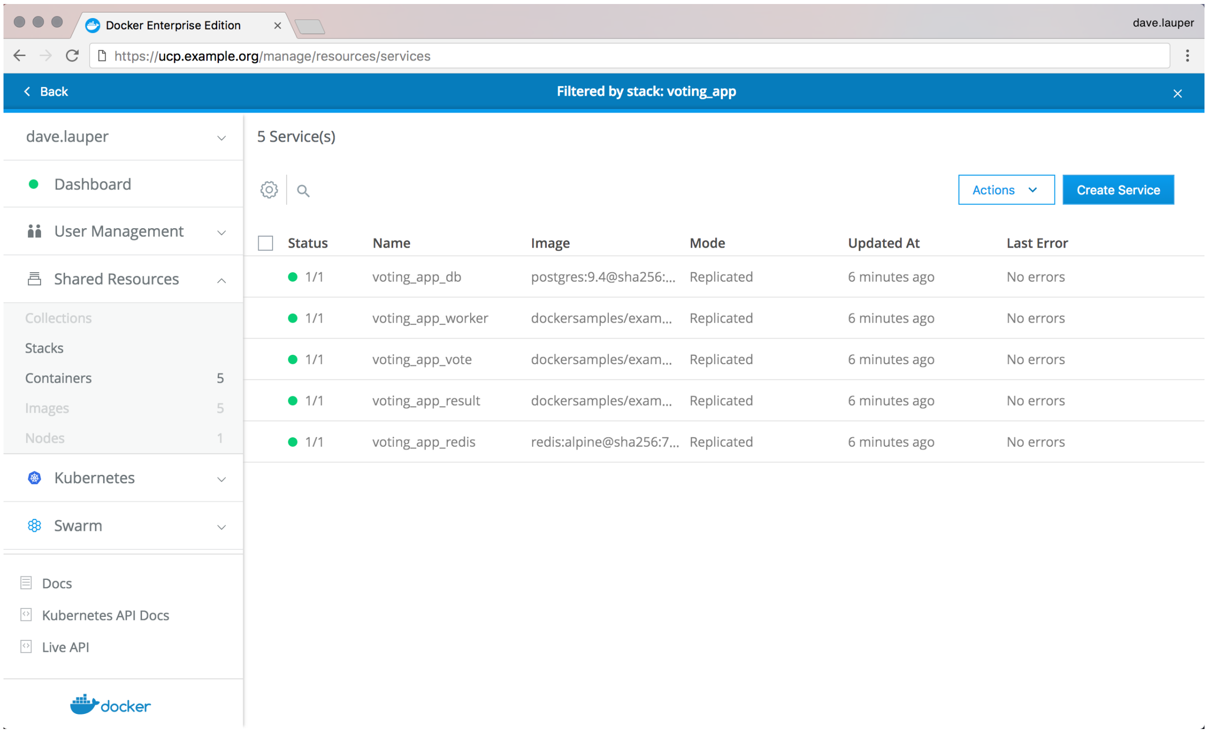
```
version: "3"
services:

  # A Redis key-value store to serve as message queue
  redis:
    image: redis:alpine
    ports:
      - "6379"
    networks:
      - frontend

  # A PostgreSQL database for persistent storage
  db:
    image: postgres:9.4
    volumes:
      - db-data:/var/lib/postgresql/data
    networks:
      - backend

  # Web UI for voting
  vote:
    image: dockersamples/examplevotingapp_vote:before
```

After that, you can see it running, and can change any aspect of the application you choose, or even edit the Compose file itself.



5 Service(s)

Status	Name	Image	Mode	Updated At	Last Error
● 1/1	voting_app_db	postgres:9.4@sha256:...	Replicated	6 minutes ago	No errors
● 1/1	voting_app_worker	dockersamples/exam...	Replicated	6 minutes ago	No errors
● 1/1	voting_app_vote	dockersamples/exam...	Replicated	6 minutes ago	No errors
● 1/1	voting_app_result	dockersamples/exam...	Replicated	6 minutes ago	No errors
● 1/1	voting_app_redis	redis:alpine@sha256:7...	Replicated	6 minutes ago	No errors

Congratulations!

You’ve taken a full-stack, dev-to-deploy tour of the entire Docker platform.

There is much more to the Docker platform than what was covered here, but you have a good idea of the basics of containers, images, services, swarms, stacks, scaling, load-balancing, volumes, and placement constraints.

Want to go deeper? Here are some resources we recommend:

- Samples (https://docs.docker.com/samples/): Our samples include multiple examples of popular software running in containers, and some good labs that teach best practices.
- User Guide (https://docs.docker.com/engine/userguide/): The user guide has several examples that explain networking and storage in greater depth than was covered here.
- Admin Guide (https://docs.docker.com/engine/admin/): Covers how to manage a Dockerized production environment.
- Training (https://training.docker.com/): Official Docker courses that offer in-person instruction and virtual classroom environments.
- Blog (https://blog.docker.com): Covers what's going on with Docker lately.

deploy (https://docs.docker.com/glossary/?term=deploy), production (https://docs.docker.com/glossary/?term=production), datacenter (https://docs.docker.com/glossary/?term=datacenter), cloud (https://docs.docker.com/glossary/?term=cloud), aws (https://docs.docker.com/glossary/?term=aws), azure (https://docs.docker.com/glossary/?term=azure), provider (https://docs.docker.com/glossary/?term=provider), admin (https://docs.docker.com/glossary/?term=admin), enterprise (https://docs.docker.com/glossary/?term=enterprise)