Docker

Development to production: Things to watch out for

* Bind mounts shouldn’t be used in production!
* Containerized apps might need a build step(e.g React apps)
* Multi-Container projects might need to be split (or should be split)across multip,e hosts/remote machines
* Trade-offs between control and responsibility be worth it!

Docker File Example

FROM node:14-alpine

WORKDIR /app

COPY pakages.json .

RUN npm install

COPY . .

EXPOSE 80

CMD [“node” , “app.js”]

Build Docker file

**docker build -t node -dep-example .**

Run containers

docker run -d --rm --name node-dep -p 80:80 node-dep-example

Installing Docker remote machine AWS

* Sudo yum update -y
* Sudo amazon-linux-extras install docker
* Sudo service docker start
* Docker run -> checking docker available or not

node-dep-example1

* sudo docker run -d --rm -p 80:80 hiteshbhatt

07/node-eample1

Upload docker hub images on EC2 instance and run with HTTP permission in security groups

How to update the remote server and push new code on it

=> docker build -t node-dep-example1 . (build updated images)

=> docker tag node-dep-example1 hiteshbhatt07/node-eample1 (tag name as per as docker images repository

=>push images in docker hub -> docker push hiteshbhatt07/node-eample1

Connect EC2 Terminal and run below command

* check container information -> docker ps
* stop decker Container by name -> docker stop nervous\_cohen
* Run docker new images which we update on docker hub
  + docker run -d --rm -p 80:80 hiteshbhatt07/node-eample1
* pull docker latest version images
  + docker pull hiteshbhatt07/node-eample1

Docker Core Concepts

* Containers
  + Isolated
  + Single-task-focused
  + Shareable, reproducible
  + Stateless(+ volumes)
* Images
  + Blueprints for Containers
  + Code + environment
  + Read-only / does not run
  + Can be built +shared
* Command
  + Build an image based on dockerfile
    - Docker build -t NAME:TAG [build image]
    - Docker run –name NAME –rm -d IAMGENAME [Run container base on image file ]
  + Share (push) an image to a registry(default: Dockerhub)
    - Docker push REPOSITORY/NAME:TAG
  + Fetch(pull)an image from a Registry
    - Docker pull REPOSITORY/NAME:TAG

Docker Container & data

Container are isolated and stateless

Kubernetes

Kubernetes is an open-source container deployment and management platform. It offers container orchestration a container runtime, and container-centric infrastructure orchestration, load balancing, self-healing mechanisms, and service discovery Kubernetes architecture ,also sometimes called Kubernetes

Command